



ISSN: 1646-9895

Revista Ibérica de Sistemas e Tecnologias de Informação  
Revista Ibérica de Sistemas y Tecnologías de Información

M a r ç o 2 4 • M a r c h 2 4



©RISTI 2024 <http://www.risti.xyz>

Nº 53

### **Edição / Edición**

Nº 53, 03/2024

**Tiragem / Tirage:** 1000

**Preço por número / Precio por número:** 17,5€

**Subscrição anual / Suscripción anual:** 30€ (2 números)

**ISSN:** 1646-9895

**Depósito legal:**

### **Indexação / Indexación**

Academic Journals Database, CiteFactor, Dialnet, DOAJ, DOI, EBSCO, GALE, IndexCopernicus, Index of Information Systems Journals, ISI Web of Knowledge, Latindex, ProQuest, QUALIS, SciELO, SCImago, Scopus, SIS, Ulrich's.

### **Publicação / Publicación**

RISTI – Revista Ibérica de Sistemas e Tecnologias de Informação

Pc. 9 de Abril, 26, 4200-422 Porto, Portugal

**E-mail:** aistic@gmail.com

**Web:** <http://www.risti.xyz>

**Director**

Álvaro Rocha, ISEG, Universidade de Lisboa, PT

**Coordenadores da Edição / Coordinadores de la Edición**

Jezreel Mejía, CIMAT, MX

**Conselho Editorial / Consejo Editorial**

Abel Méndez, Instituto Tecnológico de Costa Rica, CR

Alejandro Peña, Universidad EAFIT, CO

Alma Gómez-Rodríguez, Universidade de Vigo, ES

Ana Rita Calvão, ESTGA, Universidade de Aveiro, PT

Ania Cravero, Universidad de La Frontera, CL

António Abreu Silva, ISCAP, Instituto Politécnico do Porto, PT

Antonio Garcia, Universidad de Santiago de Compostela, ES

António Godinho, ISEC, Instituto Politécnico de Coimbra, PT

Antonio Jiménez-Martín, Universidad Politécnica de Madrid, ES

Arturo J. Méndez, Universidad de Vigo, ES

August Climent, Ramon LLull University, ES

Beatriz Rodríguez, Universidad de la Republica, UY

Borja Bordel, Universidad Politécnica de Madrid, ES

Brenda L. Flores-Rios, Universidad Autónoma de Baja California, MX

Bruno Miguel Ferreira Gonçalves, Instituto Politécnico de Bragança, PT

Carlos Alexandre Silva, Instituto Federal de Minas Gerais, BR

Carlos Carreto, Instituto Politécnico da Guarda, PT

Carlos Morais, Instituto Politécnico de Bragança, PT

Carlos Rompante Cunha, UNIAG & CeDRI & Instituto Politécnico Bragança, PT

Carlos Vaz de Carvalho, Instituto Politécnico do Porto, PT

Célio Marques, Instituto Politécnico de Tomar, PT

Ciro Martins, Universidade do Algarve, PT

Cristina Caridade, Instituto Politécnico de Coimbra, PT

Daniel Polónia, Universidade de Aveiro, PT

Dante Carrizo, Universidad de Atacama, CL

Edwin Cedeño Herrera, Universidad de Panamá, PA

Fábio Marques, ESTGA, Universidade de Aveiro, PT  
Felipe Vasquez, Universidad de La Frontera, CL  
Fernando Moreira, REMIT, IJP, Universidade Portucalense &  
IEETA, Universidade de Aveiro, PT  
Fernando Paulo Belfo, Coimbra Business School –  
ISCAC, Polytechnic Institute of Coimbra, PT  
Fernando Ribeiro, Instituto Politécnico de Castelo Branco, PT  
Fernando Bandeira, PT  
Filipe Cardoso, Instituto Politécnico de Viseu, PT  
Flor Gomes de María Sánchez Aguirre, Universidad César Vallejo, PE  
Francisco Javier Lena-Acebo, Universidad de Cantabria, ES  
Gabriel Guerrero-Contreras, Universidad de Cádiz, ES  
Gerardo González Filgueira, Universidade da Coruña, ES  
Gloria Valencia, Universidad de las Fuerzas Armadas, EC  
Guillllhermo Davila, Universidad de Lima, PE  
Hélder Gomes, Universidade de Aveiro, PT  
Helder Zagalo, Universidade de Aveiro, PT  
Hélia Guerra, Universidade dos Açores, PT  
Henrique Gil, Instituto Politécnico de Castelo Branco, PT  
Henrique S. Mamede, Universidade Aberta, Portugal  
Inês Domingues, ISEC, Instituto Politécnico de Coimbra, PT  
Isaias Bianchi, Universidade Federal de Santa Catarina, BR  
Isidro Navarro, Universidad Politécnica de Cataluña, ES  
Ismael Etxeberria-Agiriano, Universidad del País Vasco, ES  
Ivan Garcia, Universidad Tecnológica de la Mixteca, MX  
Jeimy Cano, Universidad de los Andes, CO  
João Emílio Almeida, ISTECS - Porto, PT  
João Paulo Ferreira, Instituto Politécnico de Coimbra, PT  
Joao Roberto de Toledo Quadros, CEFET/RJ, BR  
Joao Tavares, Universidade do Porto, PT  
Joaquim Reis, ISCTE - Instituto Universitário de Lisboa, PT  
Jorge Eduardo Ibarra Esque, Universidad Autónoma de Baja California, MX  
Jorge Hochstetter, Universidad de La Frontera, CL

José Álvarez-García, Universidad de Extremadura, ES  
Jose Felipe Cocon Juarez, Universidad Autónoma del Carmen, MX  
José Lousado, Instituto Politécnico de Viseu, PT  
José Luis Pastrana Brincones, Universidad de Málaga, ES  
Jose M. Molina, Universidad Carlos III de Madrid, ES  
José Ribeiro, Politécnico de Leiria, PT  
Jose Silvestre Silva, Academia Militar, PT  
Josep Maria Marco-Simó, Universitat Oberta de Catalunya, ES  
Juan Angel Contreras Vas, Universidad de Extremadura, ES  
Juan M. Santos-Gago, Universidad de Vigo, ES  
Juan Pablo DAmato, Universidad Nacional del Centro de la  
Provincia de Buenos Aires, AR  
Leila Weitzel, Universidade Federal Fluminense, BR  
Leonardo Bermón Angarita, Universidad Nacional de Colombia, CO  
Lilia Muñoz, Universidad Tecnológica de Panamá, PA  
Lucila Romero, Universidad Nacional del Litoral, AR  
Luis Álvarez Sabucedo, Universidad de Vigo, ES  
Luis Enrique Sánchez Crespo, Universidad de Castilla-la Mancha, ES  
Luisa María Romero-Moreno, Universidad Sevilla, ES  
Luz María Hernández Cruz, Universidad Autónoma de Campeche, MX  
Luz Sussy Bayona Oré, Universidad Nacional Mayor de San Carlos, PE  
Marcelo Marciszack, Universidad Tecnológica Nacional, AR  
Marcelo Zambrano Vizuite, Universidad Tecnica Del Norte  
Marco Painho, NOVA IMS, PT  
Margarita Ramirez Ramirez, Universidad Autónoma de Baja California, MX  
Maria Cristina Marcelino Bento, UNIFATEA, BR  
María de la Cruz del Río-Rama, Universidad de Vigo, ES  
Maria de los Milagros Gutierrez, Universidad Tecnológica Nacional, AR  
Maria do Rosário Bernardo, Universidade Aberta, PT  
Maria Hallo, Escola Politécnica Nacional, EC  
Maria João Ferreira, Universidade Portucalense, PT  
Maria João Gomes, Universidade do Minho, PT  
Maria Sousa, ISCTE - Instituto Universitário de Lisboa, PT

Marisol B. Correia, ESGHT - Universidade do Algarve & CiTUR, PT  
Maristela Holanda, Universidade de Brasília, BR  
Martin Llamas Nistal, Universidad de Vigo, ES  
Miguel Casquilho, Universidade de Lisboa, PT  
Miguel Ramón González Castro, Centro Tecnológico Aimen, ES  
Mirna Ariadna Muñoz Mata, CIMAT, MX  
Nelson Rocha, Universidade de Aveiro, PT  
Nuno Melão, Instituto Politécnico de Viseu, PT  
Nuno Ribeiro, Universidade Fernando Pessoa, Portugal  
Patricia Dias, Universidade do Estado de Minas Gerais, BR  
Paula Prata, Universidade da Beira Interior, PT  
Paulo Martins, Universidade de Trás-os-Montes e Alto Douro, PT  
Paulo Pinto, Universidade Nova de Lisboa, PT  
Paulo Rurato, Universidade Fernando Pessoa, Portugal  
Pedro Araújo, Universidade da Beira Interior, PT  
Pedro R. Palos- Sanchez, Universidad de Sevilla, ES  
Pedro Sanz-Angulo, Universidad de Valladolid, ES  
Pedro Sobral, Universidade Fernando Pessoa, Portugal  
Pedro Sousa, Universidade do Minho, Portugal  
René Faruk Garzozzi-Pincay, Universidad Estatal Península de Santa Elena, EC  
Ruben Pereira, ISCTE – Instituto Universitário de Lisboa, PT  
Rui Pedro Marques, Universidade de Aveiro, PT  
Rui Silva Moreira, Universidade Fernando Pessoa, PT  
Samuel Sepúlveda, Universidad de La Frontera, CL  
Santiago Raul Gonzales Sanches, Universidad Cesar Vallejo, PE  
Sara Balderas-Díaz, Universidad de Cádiz, ES  
Sergio Araya Guzmán, Universidad del Bío-Bío, CL  
Sergio F. Lopes, Universidade do Minho, PT  
Sergio Gálvez Rojas, Universidad de Málaga, ES  
Solange N Alves-Souza, Universidade de São Paulo, BR  
Telmo Silva, DigiMedia, Universidade de Aveiro, PT  
Teresa Guarda, Universidad Estatal Península de Santa Elena, EC  
Thiago Dias, Centro Federal de Educação Tecnológica de Minas Gerais, BR

Verónica Vasconcelos, Polytechnic Institute of Coimbra, PT  
Vitor Carvalho, Instituto Politécnico do Cávado e do Ave, PT  
Vitor Santos, Universidade Nova de Lisboa - NOVA IMS, PT  
Wagner Tanaka Botelho, Universidade Federal do ABC, BR  
Yamid Hernández Julio, Universidad del Sinú, CO

## Índice / Index

### EDITORIAL

|                                                                                                                                                  |   |
|--------------------------------------------------------------------------------------------------------------------------------------------------|---|
| Nuevos desafíos en Tecnologías de Información y Comunicación.....                                                                                | 1 |
| <i>Jezreel Mejía<sup>1</sup>, Mirna Muñoz, Lizbeth Alejandra Hernández-González,<br/>María del Pilar Salas-Zárate, Luis Omar Colombo Mendoza</i> |   |

### ARTIGOS / ARTICULOS / ARTICLES

|                                                                                |   |
|--------------------------------------------------------------------------------|---|
| Estandarización y Continuidad: El Puente entre ISO/IEC 29110 y DevOps .....    | 5 |
| <i>Galván-Cruz Sergio, Muñoz Mirna, Acevedo Daniela, Pastrana P. Manuel A.</i> |   |

|                                                  |    |
|--------------------------------------------------|----|
| Estudio Terciario sobre Adopción de DevOps ..... | 23 |
| <i>Brian Pando, Abel Silva, Abraham Dávila</i>   |    |

|                                                                                              |    |
|----------------------------------------------------------------------------------------------|----|
| DevOps y la Medición de la Calidad del<br>Producto de Software: Hallazgos Preliminares ..... | 37 |
| <i>Gabriel Alberto García-Mireles, Nerina Peña Olivero, Himer Avila-George</i>               |    |

|                                                                                                                                       |    |
|---------------------------------------------------------------------------------------------------------------------------------------|----|
| Factores Sociales y Culturales que Influyen en las Técnicas de<br>Priorización de Requisitos de Software: Un Estudio Secundario ..... | 53 |
| <i>Raúl Antonio Aguilar Vera, Riad Aissrad Santos Charuf,<br/>Julio C. Díaz Mendoza, Juan Pablo Ucán Pech</i>                         |    |

|                                                                                                 |    |
|-------------------------------------------------------------------------------------------------|----|
| Auditoría de riesgos de ciberseguridad: Revisión de<br>Literatura, propuesta y aplicación ..... | 69 |
| <i>I.D. Sanchez-Garcia, A.M. Rea-Guaman, T. San Feliu<sup>1</sup> and J.A. Calvo-Manzano</i>    |    |

|                                                                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Modelos de identificación de enfermedades<br>cardiovasculares implementando técnicas de<br>aprendizaje máquina: una revisión sistemática de la literatura ..... | 87 |
| <i>Johan Mardini - Bovea, Dixon Salcedo, Emiro De-la-Hoz-Franco,<br/>Yadira Quiñonez, Luz Jimenez-Roa, Víctor De la Hoz-Avila, Jhon González Sanchez</i>        |    |

|                                                                                                                |     |
|----------------------------------------------------------------------------------------------------------------|-----|
| Internet de las cosas aplicado a la agricultura:<br>estado actual y su aplicación mediante un prototipo .....  | 106 |
| <i>Jorge Luis Diaz Martinez, Dixon Salcedo, Teobaldis Mercado,<br/>Yadira Quiñonez, Andrés Mejía de la Hoz</i> |     |

|                                                                                                                                               |     |
|-----------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Propuesta de unificación de herramientas<br>bioinformáticas para el análisis de secuencias<br>proteómicas y genómicas de algas y hongos ..... | 122 |
| <i>Victor Terrón-Macias, Jezreel Mejía</i>                                                                                                    |     |



## **Nuevos desafíos en Tecnologías de Información y Comunicación**

### ***New challenges in Information and Communication Technologies***

Jezreel Mejía<sup>1</sup>, Mirna Muñoz<sup>1</sup>, Lizbeth Alejandra Hernández-González<sup>2</sup>, María del Pilar Salas-Zárate<sup>3</sup>, Luis Omar Colombo Mendoza<sup>3</sup>

**jmejia@cimat.mx; mirna.munoz@cimat.mx; lizhernandez@uv.mx;  
maria.sz@teziutlan.tecnm.mx; luis.cm@teziutlan.tecnm.mx**

<sup>1</sup>Centro de Investigación en Matemáticas/Unidad zacatecas, México.

<sup>2</sup>Universidad Veracruzana, Xalapa, Veracruz, México.

<sup>3</sup>Tecnológico Nacional de México/I.T.S. Teziutlán, Puebla, México.

**DOI: 10.17013/risti.53.1-4**

#### **1. Introducción**

Los artículos presentados en este número fueron seleccionados de los artículos aceptados en el 12vo., Congreso Internacional de Mejora de Procesos Software (CIMPS) 2023, con sede local en la Ciudad de Cuernavaca, Morelos, México del 18 al 20 de octubre del 2023. Este congreso ha tenido una tasa de aceptación del 30% de los artículos enviados.

Los artículos publicados en este número de la Revista Ibérica de Sistemas y Tecnologías de la Información (RISTI), aborda temas relacionados a “Nuevos Desafíos en Tecnologías de Información y Comunicación”. Los artículos presentan versiones extendidas y mejoradas de las versiones originales que han sido presentados en el congreso CIMPS 2023.

#### **2. Estructura**

El primer artículo se enfoca en identificar la forma en las que los principios flexibles de DevOps pueden alinearse y, al mismo tiempo, reforzar las estructuras formales del ISO/

IEC 29110. Para lograrlo, los autores evalúan hasta qué punto DevOps puede cumplir con los requisitos del ISO y garantizar que puedan mejorar la optimización de procesos, la mejora continua y la capacidad de ser dinámicos.

El segundo artículo tiene por objetivo identificar y organizar estudios secundarios sobre la adopción de DevOps. Para ello, se siguió los pasos establecidos para un mapeo sistemático de la literatura. En la consulta a las seis bases de datos consideradas, se obtuvieron 346 artículos y luego del proceso quedaron 102 estudios secundarios seleccionados. Del análisis realizado, se tiene que los temas de las mejores prácticas y recomendaciones para la adopción de DevOps son las más abordadas. De ellas, las prácticas más reportadas son las de integración continua y entrega continua, el monitoreo y la automatización de las pruebas.

El tercer artículo al igual que el segundo se enfoca en el enfoque Devops pero desde el punto de vista de la calidad. Los autores presentan los resultados de un mapeo sistemático de literatura identificando las características de calidad del producto de software que se han estudiado en entornos DevOps. Los resultados se complementan con las iniciativas de medición del desempeño general de DevOps y aquellas que miden la calidad del producto de software. Como resultado los autores definen algunas métricas de mantenibilidad y factores contextuales por considerar en estas iniciativas.

El cuarto artículo tiene como objetivo elaborar un estado del arte sobre los factores sociales y culturales, así como las técnicas vinculadas con la tarea de priorización de requisitos en el contexto del desarrollo de software. Para ello se especificaron 4 preguntas de investigación que sirvieron para guiar una revisión sistemática de literatura con la que se seleccionaron y analizaron 54 estudios primarios obtenidos de cuatro Bases de Datos y un repositorio. Entre los hallazgos se encontraron cinco categorías de factores sociales y culturales: comunicación, entendimiento del dominio, entendimiento de los stakeholders, influencias culturales y sociales, y manejo de múltiples stakeholders.

El quinto artículo los autores llevan a cabo una exploración de una guía de auditoría de riesgos de ciberseguridad tomando como referencia objetivos comunes y guías de la auditoría de riesgos de ciberseguridad. Para ello se tomó como partida una Revisión Sistemática de Literatura (SLR) considerando los estudios de los últimos diez años (2012-2022), a partir de los cuales se identificaron 23 estudios que mencionaban objetivos y guías de auditoría de riesgos de ciberseguridad. Como resultado, los autores proponen atributos que deben ser considerados para la creación de un guía de riesgos de ciberseguridad. Además, definen una guía de auditoría de riesgos de ciberseguridad (CRAG).

En el sexto artículo se presenta una revisión sistemática de la literatura sobre el procesamiento de conjuntos de datos basados en pruebas clínicas de enfermedades cardiovasculares y técnicas de Aprendizaje Automático. En este sentido, se realizó un análisis de las diferentes variables extraídas de publicaciones de revistas indexadas en bases de datos especializadas como Scopus, Web of Science, Science Direct, Biomed y Pubmed.

El séptimo artículo presenta los resultados del diseño y la prueba de un prototipo de sistema de monitoreo de variables agrícolas, las cuales pueden almacenarse y consultarse en la nube. Los autores utilizaron sensores y sistemas de procesamiento de bajo costo

y fácilmente adaptables a entornos agrícolas. Los resultados obtenidos confirman la viabilidad de implementar estos sistemas sin recurrir a herramientas tecnológicas y computacionales complejas.

Finalmente, en el octavo artículo se propone una aplicación web que unifica diferentes funcionalidades útiles en el análisis de secuencias biológicas. Desarrollada en Python y Django, integra funcionalidades de BioPython, BioPandas, 3Dmol.js y BLAST, entre otras, que además posibilita su uso a usuarios sin conocimientos de programación. La herramienta abarca el análisis de secuencias de ADN y proteínas, transcripción, alineación, búsqueda de secuencias y más. Dentro de las características más destacables está la eficiencia, precisión al obtener datos cuantitativos del proceso que se está ejecutando y la facilidad de uso.

### **3. Agradecimientos**

Agradecemos al director de la Revista Álvaro Rocha por otorgar la oportunidad para dar a conocer los trabajos más relevantes presentados en el congreso CIMPS 2023. Asimismo, expresamos nuestra gratitud a los autores de los trabajos publicados, lo cual hizo posible este número en RISTI. Damos la enhorabuena a todos aquellos autores cuyos artículos han sido publicados en este número de RISTI.



# Estandarización y Continuidad: El Puente entre ISO/IEC 29110 y DevOps

Galván-Cruz Sergio<sup>1</sup>, Muñoz Mirna<sup>1\*</sup>, Acevedo Daniela<sup>1</sup>, Pastrana P. Manuel A.<sup>2</sup>

{daniela.acevedo; mirna.munoz; sergio.galvan}@cimat.mx;  
mapastrana@admon.uniajc.edu.co

<sup>1</sup> Centro de Investigación en Matemáticas- Unidad Zacatecas, México.

<sup>2</sup> Institución Universitaria Antonio José Camacho, Cali, Valle del Cauca, Colombia

DOI: 10.17013/risti.53.5-22

**Resumen:** Este artículo se enfoca en la integración entre DevOps, que es un conjunto de prácticas que enfatiza la agilidad, la automatización y la colaboración entre los equipos de desarrollo y operaciones, y el perfil básico del estándar ISO/IEC 29110, que proporciona pautas para usar estándares de calidad y procesos de desarrollo de software en entidades muy pequeñas (EMPs). Esta investigación se enfoca en identificar la forma en las que los principios flexibles de DevOps pueden alinearse y, al mismo tiempo, reforzar las estructuras formales del ISO/IEC 29110. Para lograrlo, se evalúa hasta qué punto DevOps puede cumplir con los requisitos del ISO y garantizar que puedan mejorar la optimización de procesos, la mejora continua y la capacidad de ser dinámicos. Finalmente, se descubrirán elementos que servirían a las EMPs para mejorar su proceso de desarrollo y permanezcan conformes con los estándares internacionales.

**Palabras-clave:** ISO/IEC 29110, DevOps, mapeo, buenas prácticas, Entidades Muy Pequeñas, EMPs

## *Standardization and Continuity: The Bridge between ISO/IEC 29110 and DevOps*

**Abstract:** This article focuses on the integration between DevOps, which is a set of practices that emphasizes agility, automation, and collaboration between development and operations teams, and the basic profile of the ISO/IEC 29110 standard, which provides guidelines for using quality standards and software development processes in Very Small Entities (VSEs). This research focuses on identifying how the flexible principles of DevOps can align and, at the same time, reinforce the formal structure of ISO/IEC 29110. To achieve this, the extent to which DevOps can meet the requirements of ISO is evaluated, and the extent to which they can improve process optimization, continuous improvement, and dynamic ability are ensured. Finally, elements will be discovered that would help EMPs improve their development process and remain in compliance with international standards.

**Keywords:** ISO/IEC 29110, DevOps, mapping, best practices, Very Small Entities, VSEs

## 1. Introducción

DevOps ha ganado popularidad, debido a su capacidad para lanzar aplicaciones al mercado de manera rápida, eficiente, con bajo costo de desarrollo y mínimo esfuerzo humano como sugieren (Erich, Amrit & Daneva, 2014; Mousaei & Gandomani 2020; Almeida, Simões & Lopes, 2022). Adicionalmente, otros autores resaltan algunos beneficios del uso de DevOps tales como: time to market mejorado, cerrar la brecha entre el equipo de desarrollo (Dev) y Operaciones (Ops), lanzamientos confiables, calidad del producto mejorada, mejora y automatización de procesos entre otros (Ghofrani & Bozorgmehr, 2019; Guerrero et al., 2020). Así mismo, el trabajo de (Khan et al., 2022) resalta que las organizaciones están adoptando masivamente DevOps en un intento de satisfacer la creciente demanda de los clientes por aplicaciones altamente disponibles, de lanzamiento continuo y de gran valor, accesibles en todo momento y lugar y en cualquier plataforma.

Sin embargo, la adopción de DevOps representa un reto considerable, resaltando entre los desafíos existentes que dificultan su implementación: 1) falta de comprensión del significado de DevOps (Kamuto & Langerman, 2017), 2) resistencia al cambio por parte de las personas (Kamuto & Langerman, 2017), 3) falta de conocimiento para la elección e integración correcta de herramientas tecnológicas (Ghantous & Gill, 2017), 4) diferencias en el estilo de trabajo entre Dev y Ops (Erich et al., 2017), 5) diseño de la arquitectura para soportar DevOps (Bucena & Kirikova, 2017), 6) cambio en los procesos (Chen, 2017), y 7) cambiar estructuras y estándares organizacionales (Bucena & Kirikova, 2017).

Los retos antes mencionados representan barreras que las empresas deben eliminar para poder realizar una implementación exitosa de DevOps, además, cabe resaltar que siete de los retos están relacionados con la inexistencia de una guía clara y detallada de sus procedimientos que brinde el soporte para reforzar la comprensión de DevOps, al mismo tiempo que ayude en la adopción de prácticas y facilite su implementación a través de la sugerencia de herramientas tecnológicas, siendo el alcance de este trabajo de investigación.

En este contexto, este artículo se centra en realizar un mapeo entre las tareas de la ISO/IEC 29110 y las actividades del proceso DevOps, como punto de partida para la creación de una guía clara que ayude a la adopción de prácticas y facilite la implementación.

El objetivo de este artículo es explorar y establecer la relación entre las prácticas de DevOps y las tareas estipuladas en el perfil básico del estándar ISO/IEC 29110.

El resto del artículo está estructurado como a continuación se menciona, la sección 2 presenta los conceptos clave en los que se basa esta investigación, la sección 3 describe la metodología seguida para realizar el mapeo, la sección 4 analiza los resultados, y finalmente, la sección 5 presenta las conclusiones y trabajo futuro.

## 2. Marco Teórico

El presente estudio se enfoca en identificar las relaciones entre DevOps, que es un conjunto de prácticas caracterizado por su agilidad, automatización y colaboración

continua entre dos áreas como lo son el desarrollo y las operaciones, y el perfil básico del estándar ISO/IEC 29110, diseñado para mejorar la calidad y el proceso sistemático en el desarrollo de software, particularmente en entidades muy pequeñas (con un máximo de 25 empleados). Esta sinergia se explora con el fin de descubrir cómo la estructura del estándar ISO/IEC 29110 puede integrarse coexistir y potenciarse con la flexibilidad de DevOps. Adicionalmente, esta investigación propone analizar los principios, prácticas y beneficios de DevOps, como la integración y entrega continua, y contrastarlos con los requisitos y procedimientos estipulados en el perfil básico del ISO/IEC 29110.

### 2.1. DevOps

De acuerdo a autores como Sharma (Sharma & Coyne, 2015), Jabbari (Jabbari et al., 2016), Lwakatare (Lwakatare et al, 2016), Ståhl & Bosch (Ståhl, D., & Bosch, 2017) y França (França, Jeronimo & Travassos, 2016) “DevOps es un proceso empresarial, que está compuesto por un conjunto de actividades o tareas que producen un resultado específico (servicio o producto) para los clientes, desde el desarrollo y comprobación hasta la producción”, pero ampliando, posterior a la puesta de producción, este proceso abarca también las operaciones y el monitoreo del sistema en producción, así como la planificación de las tareas a realizar que fomentan la mejora continua producto de la información que se ha generado en otras actividades de DevOps. La Figura 1, muestra una representación genérica del proceso de DevOps.

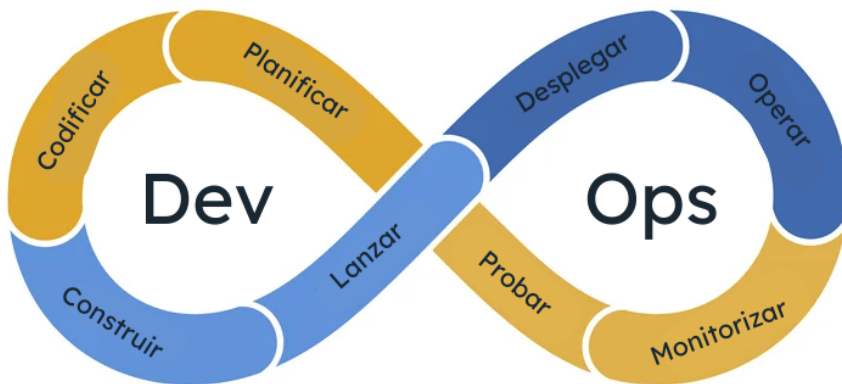


Figura 1 – Proceso DevOps (Redondo & Cárdenas, 2022)

Como se observa en la Figura 1, DevOps puede entenderse como un proceso con 8 actividades, las cuales están compuestas por un conjunto de prácticas que lo sustentan. A continuación, se definen los conceptos concernientes:

- Un proceso es un grupo de tareas y/o actividades relacionadas entre sí que persiguen cubrir las necesidades de los clientes (SOLÍS & TINOCO, 2015).
- Las actividades son un grupo de acciones que persiguen un mismo fin (SOLÍS & TINOCO, 2015).

- Una práctica es un ejercicio o realización de una acción de forma continuada y conforme a sus reglas (Lawrence Pfleeger, 2002).

De acuerdo con (Dominguez-Acosta & Garcia-Mireles, 2021; López Peña, 2021; Perera et al., 2017), DevOps está compuesto por las siguientes actividades:

- *Planificación:* Los equipos de DevOps se encargan de conceptualizar y diseñar el software que van a desarrollar, identificando las necesidades del cliente y proponiendo soluciones a través de la elaboración o mejora de un software. En esta actividad se detallan las funcionalidades y características que tendrá el software. Además, se planifica qué herramientas utilizar para que el equipo tenga claras las tareas pendientes, el progreso realizado, y evalúen si podrán cumplir con los plazos establecidos, entre otros aspectos.
- *Codificación:* en esta actividad se realiza el desarrollo o mejora del software a construir, teniendo en cuenta los elementos de la planeación, e implementando prácticas que garanticen la organización del código, la trazabilidad de los cambios y su alta disponibilidad para todos los integrantes del equipo en cualquier momento.
- *Construcción:* en esta actividad se realiza la compilación, enlace o términos similares referentes a crear el empaquetado de la solución. Esta actividad depende de las instrucciones propias del lenguaje de programación y es una actividad repetitiva que se puede automatizar.
- *Pruebas:* esta actividad busca comprobar que la calidad de lo construido sea la más alta, centrado su atención no solamente en comprobar que la funcionalidad responde correctamente a la descripción funcional del análisis, además permite determinar cómo responde a distintos estímulos que producirán un error.
- *Liberación:* en esta actividad se automatizan los pasos anteriores encadenando varias prácticas, lo que permite a partir de los cambios registrados en la codificación, iniciar la construcción, pruebas y crear así un entregable listo para ser puesto en un servidor.
- *Despliegue:* esta actividad se enfoca en tomar la unidad construida ya sea de forma manual o automatizada y la pone operativa en un servidor.
- *Operación:* esta actividad provee un conjunto de prácticas que garantizan la estabilidad, replicabilidad y recuperación de fallos rápida, en la infraestructura que soporta los diversos aplicativos y servicios de la empresa.
- *Monitoreo:* esta actividad provee a través de varias prácticas la generación de métricas que permiten estar pendiente del comportamiento y las variaciones que tengan tanto los aplicativos como la infraestructura, con el ánimo de fomentar la mejora continua.

## 2.2.ISO/IEC 29110

De acuerdo con (Bolívar et al., 2023; Galvan-Cruz et al., 2021; Muñoz, 2022) el estándar ISO/IEC 29110, titulado “Ingeniería de Software y Sistemas - Perfiles de Ciclo de Vida para Entidades Muy Pequeñas (VSE)”, surge como respuesta a la

necesidad de simplificar los procesos de desarrollo y gestión de proyectos para empresas y organizaciones que desarrollan software, y que cuentan con un máximo de 25 empleados. Este estándar fue elaborado por el Grupo de Trabajo 24 (WG24) del Subcomité 7 (SC7) del Comité Técnico Conjunto 1 (JTC1) de la ISO en 2005. El estándar tiene como propósito general proporcionar un marco de proceso simple y práctico para la ejecución de proyectos de software. Este estándar abarca 2 procesos: Gestión de Proyectos (GP) e Implementación de Software (IS). A continuación, se detalla cada uno:

**2.2.1. Procesos del estándar ISO/IEC 29110**

Los procesos que componen el perfil básico del estándar ISO/IEC 29110 están enfocados en proporcionar una guía para la gestión de proyectos y la implementación técnica en pequeñas organizaciones (O'Connor & Laporte, 2017; Moreno-Campos et al., 2014; Takeuchi et al., 2014).

*Gestión de Proyectos*

El propósito del proceso de GP es establecer y realizar de manera sistemática las tareas del proyecto de implementación de software, lo que permite cumplir con los objetivos del proyecto en la calidad, tiempo y costos esperados (ISO/IEC, 2011). El proceso de GP contiene 4 actividades y sus respectivas tareas. La Tabla 1 lista las actividades y número de tareas asociadas.

| Nombre de la Actividad |                                   | Cantidad de tareas |
|------------------------|-----------------------------------|--------------------|
| GP.1                   | Planificación del Proyecto        | 15                 |
| GP.2                   | Ejecución del Plan del Proyecto   | 6                  |
| GP.3                   | Evaluación y control del proyecto | 3                  |
| GP.4                   | Cierre del proyecto               | 2                  |

Tabla 1 – Muestra las actividades del proceso GP del ISO/IEC 29110 perfil básico

*Implementación de software*

El propósito del proceso de IS es la realización sistemática de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software nuevos o modificados de acuerdo con los requisitos especificados (ISO/IEC, 2011). El proceso de GP se contiene 4 actividades y sus respectivas tareas. La Tabla 2 lista las actividades y número de tareas asociadas.

| Nombre de la Actividad |                                          | Cantidad de tareas |
|------------------------|------------------------------------------|--------------------|
| IS.1                   | Inicio de la Implementación del software | 2                  |
| IS.2                   | Análisis de requisitos de software       | 7                  |

|      | Nombre de la Actividad                      | Cantidad de tareas |
|------|---------------------------------------------|--------------------|
| IS.3 | Arquitectura y diseño detallado de software | 8                  |
| IS.4 | Construcción del software                   | 7                  |
| IS.5 | Integración y pruebas de software           | 11                 |
| IS.6 | Entrega del producto                        | 6                  |

Tabla 2 – Muestra las actividades del proceso de Implementación de Software del ISO/IEC 29110 perfil básico

### 3. Metodología

La metodología definida para la realización del mapeo comprendió un enfoque estructurado, el cual se detalla a continuación:

- a. Determinar el propósito y los objetivos de las actividades del proceso DevOps, desglosado en dos fases fundamentales:
  - Investigar y analizar fuentes bibliográficas relevantes, incluyendo artículos especializados y textos sobre DevOps, para obtener una comprensión profunda de la materia.
  - Sintetizar las actividades de DevOps, estableciendo claramente su propósito y los objetivos perseguidos, para delineación de su valor y aplicaciones.
- b. Elaborar una matriz de correspondencia entre las tareas especificadas en el perfil básico del estándar ISO/IEC 29110 y las actividades inherentes al proceso de DevOps, buscando alinear los estándares de calidad y las prácticas de eficiencia operativa.
  - Este instrumento clave ha sido cuidadosamente examinado y validado por expertos en el ámbito de la ingeniería de software y especialistas en el estándar ISO/IEC 29110 y en el proceso DevOps. El consenso de los expertos ha permitido refinar la matriz para que sirva como un estudio confiable alineados a los principios de DevOps y los procesos definidos por el estándar ISO/IEC 29110.
- c. Presentar los resultados, donde se muestra la convergencia entre el perfil básico del estándar ISO/IEC 29110 y el proceso DevOps, destacando hallazgos clave, proyectando implicaciones prácticas y teóricas para el campo de la ingeniería de software.

### 4. Análisis y Experimentación

La matriz de correspondencia diseñada en este estudio constituye un esfuerzo analítico y sistematizado para tender puentes entre el estructurado entorno del estándar ISO/IEC 29110 y las dinámicas prácticas de DevOps. A través de esta matriz, cada tarea detallada dentro de los procesos de gestión de proyectos y de implementación de software del perfil

básico de ISO/IEC 29110 se mapea meticulosamente con las actividades de DevOps, incluyendo la integración y despliegue continuos, la automatización, el monitoreo y la gestión de infraestructura.

El propósito de este mapeo es realizar una comparativa para clarificar cómo el estándar y las prácticas de DevOps pueden complementar, enriquecer o, en algunos casos, mejorar optimizar las tareas de estándar, proporcionando así una base formal para las organizaciones que buscan armonizar estos dos mundos en su práctica de Ingeniería de Software.

Para cada una de las matrices de correspondencia mostradas en las siguientes secciones se muestra la tarea de cada actividad del ISO/IEC 29110 y su relación con las actividades y prácticas DevOps, en caso de existir alguna relación se marca con una X de lo contrario se mantiene en blanco, lo que significa que no se encontró valor alguno.

| ISO/IEC 29110 |                                                                                                                                                                                                                                                                                                                                                                                                                                         | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                                                                                                                                                                                                                                                                                 | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 1.1           | Revisar el enunciado de trabajo                                                                                                                                                                                                                                                                                                                                                                                                         | X                  |              |              |         |            |            |           |           |
| 1.2           | Definir con el Cliente las Instrucciones de Entrega de cada uno de los Entregables especificados en el Enunciado de Trabajo                                                                                                                                                                                                                                                                                                             | X                  |              |              |         |            |            |           |           |
| 1.3           | Identificar las Tareas específicas a realizar para producir los Entregables y sus Componentes de Software identificados en el Enunciado de Trabajo. Incluir Tareas en el proceso IS junto con la verificación, validación y revisiones con el Cliente y las Tareas del Equipo de Trabajo para asegurar la calidad de los productos de trabajo. Identificar las Tareas para realizar las Instrucciones de Entrega. Documentar las Tareas | X                  |              |              |         |            |            |           |           |
| 1.4           | Establecer la Duración Estimada para realizar cada tarea                                                                                                                                                                                                                                                                                                                                                                                | X                  |              |              |         |            |            |           | X         |
| 1.5           | Identificar y documentar los Recursos: humanos, materiales, equipos y herramientas, normas, incluyendo la formación requerida del Equipo de Trabajo para realizar el proyecto. Incluir en el cronograma las fechas en que se necesitarán los Recursos y la formación                                                                                                                                                                    | X                  | X            | X            | X       | X          | X          | X         |           |
| 1.6           | Establecer la Composición del Equipo de Trabajo asignando roles y responsabilidades según los Recursos                                                                                                                                                                                                                                                                                                                                  | X                  |              |              |         |            |            |           |           |
| 1.7           | Asignar las fechas estimadas de inicio y finalización a cada una de las Tareas para crear el Cronograma de las Tareas del Proyecto teniendo en cuenta los Recursos asignados, la secuencia y dependencia de las Tareas                                                                                                                                                                                                                  | X                  |              |              |         |            |            |           |           |
| 1.8           | Calcular y documentar el Esfuerzo y Costo Estimados del proyecto                                                                                                                                                                                                                                                                                                                                                                        | X                  |              |              |         |            |            |           | X         |
| 1.9           | Identificar y documentar los riesgos que pueden afectar al proyecto                                                                                                                                                                                                                                                                                                                                                                     | X                  | X            | X            | X       | X          | X          | X         |           |
| 1.10          | Documentar la Estrategia de Control de Versiones en el Plan del Proyecto                                                                                                                                                                                                                                                                                                                                                                | X                  | X            |              |         |            |            |           |           |
| 1.11          | Generar el Plan del Proyecto integrando los elementos previamente identificados y documentados                                                                                                                                                                                                                                                                                                                                          | X                  |              |              |         |            |            |           |           |
| 1.12          | Incluir la Descripción del Producto, Alcance, Objetivos y Entregables en el Plan del Proyecto                                                                                                                                                                                                                                                                                                                                           | X                  |              |              |         |            |            |           |           |
| 1.13          | Verificar y obtener la aprobación del Plan del Proyecto                                                                                                                                                                                                                                                                                                                                                                                 | X                  |              |              |         |            |            |           |           |
| 1.14          | Revisar y aceptar el Plan del Proyecto                                                                                                                                                                                                                                                                                                                                                                                                  | X                  |              |              |         |            |            |           |           |
| 1.15          | Establecer el Repositorio del Proyecto utilizando la Estrategia de Control de Versiones                                                                                                                                                                                                                                                                                                                                                 | X                  | X            |              |         |            |            |           |           |

Tabla 3 – Tabla de Correspondencia de la actividad GP Planificación del Proyecto respecto a DevOps

Planificación del Proyecto

La planificación de proyectos según el estándar ISO/IEC 29110 y las actividades de DevOps se entrelazan armónicamente, creando un ecosistema donde la metodología estructurada de ISO fomenta una base sólida para las fases de planificación, desarrollo y pruebas, mientras que DevOps inyecta agilidad y mejora continua en el ciclo de vida del software con prácticas como integración continua, entrega continua, despliegue automatizado y monitoreo constante. Esta relación sinérgica permite que los proyectos se beneficien de la previsión y claridad de ISO, complementándose con la eficiencia y respuesta rápida de DevOps para adaptarse a los cambios y mejorar la calidad del software entregado (ISO/IEC, 2011).

Ejecución del plan de proyecto

La ejecución del plan de proyecto según el ISO/IEC 29110 se alinea estrechamente con las actividades dinámicas de DevOps. La meticulosa planificación ISO/IEC 29110 establece un camino claro que DevOps sigue con su ciclo de retroalimentación rápida en las etapas de codificación y construcción. Adicionalmente, la rigurosidad en las pruebas de ISO/IEC se refleja en la automatización y la entrega continua de DevOps. Además, las etapas de liberación y despliegue se transforman mediante la implementación automatizada de DevOps; y finalmente, la operación y el monitoreo, siempre presentes en ISO/IEC, se intensifican en DevOps para un ajuste continuo y optimización del rendimiento, lo que demuestra una convergencia complementaria de disciplina y adaptabilidad entre ambas metodologías (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                       | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                               | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 2.1           | Monitorear la ejecución del Plan de Proyecto y registrar los datos reales en el Registro de Estado de Progreso.                                                       | X                  | X            | X            | X       | X          | X          | X         | X         |
| 2.2           | Analizar y evaluar la Solicitud de Cambio por costo, programación e impacto técnico.                                                                                  | X                  | X            | X            | X       | X          | X          | X         | X         |
| 2.3           | Realizar reuniones de revisión con el Equipo de Trabajo, identificar problemas, revisar el estado de riesgo, registrar acuerdos y darles seguimiento hasta su cierre. | X                  | X            | X            | X       | X          | X          | X         | X         |
| 2.4           | Realizar reuniones de revisión con el Cliente, registrar acuerdos y darles seguimiento hasta su cierre.                                                               | X                  |              |              |         |            |            |           | X         |
| 2.5           | Realizar copias de seguridad de acuerdo con la Estrategia de Control de Versiones.                                                                                    | X                  | X            | X            | X       | X          | X          | X         | X         |
| 2.6           | Realizar la recuperación del Repositorio del Proyecto utilizando la Copia de Seguridad del Repositorio del Proyecto, si es necesario.                                 | X                  | X            | X            | X       | X          | X          | X         |           |

Tabla 4 – Tabla de Correspondencia de la actividad GP Ejecución del Plan del Proyecto respecto a DevOps

Evaluación y Control del proyecto

En el marco de la evaluación y control del proyecto del ISO/IEC 29110, la estructura y medición detalladas del avance del proyecto complementan la naturaleza cíclica de DevOps, en la que la planificación es iterativa, la codificación y construcción son continuas, y las pruebas son automáticas y frecuentes. La liberación y el despliegue en DevOps, ágiles y automatizados, se benefician de la precisión de seguimiento de ISO, mientras que la operación y el monitoreo en DevOps proporcionan datos en tiempo real que alimentan de vuelta al sistema de control del ISO, permitiendo así una mejora y adaptación continuas dentro de un marco de trabajo controlado y sistemático (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                                                                                                                                                                                                                                                                                        | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                                                                                                                                                                                                                                                                                | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 3.1           | Evaluación del progreso del proyecto con respecto al Plan del Proyecto, comparando:<br>-Tareas reales frente a Tareas planificadas<br>-resultados reales frente a objetivos del proyecto establecidos<br>-asignación de recursos reales frente a Recursos planificados<br>-coste real frente a estimaciones presupuestarias<br>-tiempo real frente a programación planificada<br>-riesgo real frente a riesgo previamente identificado | X                  |              |              |         |            |            |           | X         |
| 3.2           | Establecer acciones para corregir desviaciones o problemas y riesgos identificados que afecten al cumplimiento del plan, según sea necesario, documentarlos en el Registro de Correcciones y hacer seguimiento hasta su cierre.                                                                                                                                                                                                        | X                  | X            | X            | X       | X          | X          | X         | X         |
| 3.3           | Identificar cambios en los requisitos y/o en el Plan del Proyecto para abordar desviaciones importantes, riesgos potenciales o problemas relacionados con la realización del plan, documentarlos en la Solicitud de Cambio y hacer seguimiento hasta su cierre.                                                                                                                                                                        | X                  | X            | X            | X       | X          | X          | X         |           |

Tabla 5 – Tabla de Correspondencia de la actividad GP Evaluación y Control del Proyecto respecto a DevOps

Cierre del Proyecto

El cierre del proyecto de acuerdo al ISO/IEC 29110, implica la conclusión formal y la reflexión sobre el proyecto, se entrelaza con DevOps a través de un enfoque de mejora continua: mientras que ISO promueve la documentación final y la revisión de los logros en comparación con los objetivos iniciales, DevOps ve el ‘cierre’ como una fase transitoria hacia el próximo ciclo de vida, donde la retroalimentación obtenida en etapas de planificación, codificación, construcción, pruebas, liberación, despliegue, operación y monitoreo informa las futuras iteraciones, garantizando que cada ‘fin’ es el precursor de nuevas mejoras y desarrollos en el flujo de trabajo del software (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                                                                       | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                                                               | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 4.1           | Formalizar la finalización del proyecto de acuerdo con las Instrucciones de Entrega establecidas en el Plan del Proyecto, proporcionando soporte para la aceptación y obteniendo la firma del Registro de Aceptación. | X                  |              |              |         | X          | X          | x         | X         |
| 4.2           | Actualizar el Repositorio del Proyecto.                                                                                                                                                                               | X                  | X            | X            | X       | X          | X          | X         | X         |

Tabla 6 – Tabla de Correspondencia de la actividad GP Cierre del Proyecto respecto a DevOps

Inicio de la Implementación del Software

El inicio de la implementación del software en el marco del ISO/IEC 29110 sienta las bases metodológicas que se integran con las prácticas de DevOps, iniciando con una planificación robusta que define los requisitos y el alcance del proyecto. Esta planificación inicial se complementa con la rápida iteración de DevOps en codificación y construcción, facilitando una transición suave a las pruebas y asegurando que la liberación y el despliegue sean ágiles y estén alineados con los objetivos iniciales. La operación y el monitoreo de DevOps, comenzando ya en las primeras fases del proyecto, permite una retroalimentación temprana y una adaptación continua, lo que refuerza el enfoque de ISO para un inicio de proyecto bien definido y controlado (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                       | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                               | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 1.1           | Revisar el Plan de Proyecto actual con los miembros del Equipo de Trabajo para lograr un entendimiento común y obtener su compromiso con el proyecto. | X                  |              |              |         |            |            |           |           |
| 1.2           | Establecer o actualizar el entorno de implementación.                                                                                                 | X                  | X            | X            | X       | X          | X          | X         |           |

Tabla 7 – Tabla de Correspondencia de la actividad IS Inicio de la Implementación de Software respecto a DevOps

Análisis de los requisitos de software

| ISO/IEC 29110 |                                                                                                                                                       | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                               | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 2.1           | Asignar Tareas a los miembros del Equipo de Trabajo de acuerdo con su rol, basándose en el Plan de Proyecto actual.                                   | X                  |              |              |         |            |            |           |           |
| 2.2           | Documentar o actualizar la Especificación de Requisitos.                                                                                              | X                  |              |              |         |            |            |           | X         |
| 2.3           | Verificar y obtener la aprobación de la Especificación de Requisitos.                                                                                 | X                  |              |              | X       |            |            |           |           |
| 2.4           | Validar y obtener la aprobación de la Especificación de Requisitos.                                                                                   | X                  |              |              |         |            |            |           |           |
| 2.5           | Documentar la versión preliminar de la *Documentación del Usuario del Software o actualizar el manual actual, si es apropiado. *(Opcional)            | X                  |              |              |         |            |            |           |           |
| 2.6           | Verificar y obtener la aprobación de la *Documentación del Usuario del Software, si es apropiado. *(Opcional)                                         | X                  |              |              |         |            |            |           |           |
| 2.7           | Incorporar la Especificación de Requisitos y la *Documentación del Usuario del Software a la Configuración del Software en la línea base. *(Opcional) | X                  |              |              |         |            |            |           |           |

Tabla 8 – Tabla de Correspondencia de la actividad IS Análisis de los Requisitos de Software respecto a DevOps

El análisis de requisitos de software del ISO/IEC 29110 establece una alineación con DevOps iniciando con la planificación detallada, que aborda los requisitos funcionales y no funcionales esenciales para la codificación y construcción del software. Esta fase

crítica en ISO se refleja en DevOps a través de prácticas como la infraestructura como código y la automatización de pruebas, que validan continuamente los requisitos a medida que el software evoluciona. Además, las actividades de liberación, despliegue, operación y monitoreo en DevOps están guiadas por los criterios establecidos durante el análisis de requisitos, asegurando que el software desarrollado cumpla con las expectativas del cliente y se mantenga eficiente y confiable en entornos de producción reales (ISO/IEC, 2011).

*Arquitectura y diseño detallado de software.*

La arquitectura y el diseño detallado de software en ISO/IEC 29110 proporcionan una hoja de ruta técnica que se alinea con DevOps, asegurando que cada actividad, desde la planificación inicial hasta la codificación y la construcción, se ejecuta con una comprensión clara de la infraestructura y los patrones de diseño del sistema. Este enfoque meticuloso permite que las prácticas de integración continua y entrega continua de DevOps se implementen eficazmente, garantizando que las pruebas, la liberación y el despliegue se realicen de manera cohesiva y en consonancia con la arquitectura prevista. Asimismo, facilita la operación y el monitoreo post-despliegue, posibilitando el monitoreo y la gestión proactiva del rendimiento basadas en la arquitectura y el diseño previamente definidos, lo que da lugar a sistemas estables y escalables en producción (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                                                                                              | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                                                                                      | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 3.1           | Asignar Tareas a los miembros del Equipo de Trabajo relacionados con su rol de acuerdo con el Plan de Proyecto actual.                                                                                                                       | X                  |              |              |         |            |            |           | X         |
| 3.2           | Entender la Especificación de Requisitos.                                                                                                                                                                                                    |                    | X            |              | X       | X          | X          | X         | X         |
| 3.3           | Documentar o actualizar el Diseño del Software.                                                                                                                                                                                              | X                  | X            | X            | X       | X          | X          | X         |           |
| 3.4           | Verificar y obtener la aprobación del Diseño del Software.                                                                                                                                                                                   | X                  |              |              |         |            |            |           |           |
| 3.5           | Establecer o actualizar los Casos de Prueba y los Procedimientos de Prueba para las pruebas de integración basados en la Especificación de Requisitos y el Diseño del Software. El cliente proporciona los datos de prueba, si es necesario. | X                  |              |              |         |            |            |           |           |
| 3.6           | Verificar y obtener la aprobación de los Casos de Prueba y los Procedimientos de Prueba.                                                                                                                                                     | X                  |              |              |         |            |            |           |           |
| 3.7           | Actualizar el Registro de Trazabilidad incorporando los Casos de Prueba y los Procedimientos de Prueba.                                                                                                                                      | X                  |              |              |         |            |            |           |           |
| 3.8           | Incorporar el Diseño del Software y el Registro de Trazabilidad a la Configuración del Software como parte de la base. Incorporar los Casos de Prueba y los Procedimientos de Prueba al Repositorio del Proyecto.                            | X                  |              |              |         |            |            |           |           |

Tabla 9 – Tabla de Correspondencia de la actividad IS de la Arquitectura y Diseño detallado de software respecto a DevOps

Construcción del software

En el ISO/IEC 29110, la construcción del software se enfoca en transformar los requisitos y el diseño en un producto operativo, resuénalo que es acorde lo cual resuena con la filosofía de DevOps al adoptar prácticas de automatización y herramientas de integración continua que facilitan la transición desde la codificación hasta la construcción y pruebas. Esta etapa de construcción se enriquece con un flujo de trabajo de DevOps que integra la liberación y el despliegue continuos, asegurando que el producto construido sea desplegado de manera eficiente en la operación. El monitoreo constante en DevOps, alimentado por la fase de construcción, permite ajustes rápidos y mejora iterativa, garantizando que el software no solo se construye para cumplir con las especificaciones iniciales, sino que también evoluciona en respuesta a los comentarios y necesidades operativas reales (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                         | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                 | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 4.1           | Asignar Tareas a los miembros del Equipo de Trabajo relacionados con su rol, de acuerdo con el Plan de Proyecto actual.                                                 | X                  |              |              |         |            |            |           | X         |
| 4.2           | Entender el Diseño del Software.                                                                                                                                        | X                  | X            |              | X       | X          | X          | X         | X         |
| 4.3           | Construir o actualizar los Componentes del Software basados en la parte detallada del Diseño del Software.                                                              | X                  | X            | X            | X       | X          | X          | X         | X         |
| 4.4           | Diseñar o actualizar los casos de prueba unitarios y aplicarlos para verificar que los Componentes del Software implementan la parte detallada del Diseño del Software. | X                  |              |              | X       |            |            |           | X         |
| 4.5           | Corregir los defectos encontrados hasta lograr una prueba unitaria exitosa (alcanzando los criterios de salida).                                                        |                    | X            | X            | X       | X          | X          | X         |           |
| 4.6           | Actualizar el Registro de Trazabilidad incorporando los Componentes del Software construidos o modificados.                                                             | x                  |              |              |         |            |            |           |           |
| 4.7           | Incorporar los Componentes del Software y el Registro de Trazabilidad a la Configuración del Software como parte de la base.                                            | X                  | X            |              |         |            |            |           |           |

Tabla 10 – Tabla de Correspondencia de la actividad IS Construcción del Software respecto a DevOps

Integración y pruebas de software

La integración y pruebas de software según el ISO/IEC 29110 se alinean estrechamente con las prácticas DevOps que priorizan la automatización de las pruebas y la integración continua para validar la calidad y funcionalidad del software a lo largo de su desarrollo. En este entorno, cada incremento de software se somete a pruebas tras su construcción, integrando cambios pequeños y frecuentes que se verifican automáticamente para garantizar su adecuación antes de avanzar a la siguiente etapa. Esta práctica no solo refuerza la confianza en la estabilidad del software, sino que también permite una liberación y despliegue más ágiles, facilitando un ciclo de retroalimentación continuo que alimenta directamente la operación y el monitoreo del software en ambientes productivos. Este enfoque integrado y fluido permite a los equipos de DevOps ajustar rápidamente las estrategias y procesos en respuesta a cualquier desafío o necesidad

emergente, optimizando el rendimiento y la fiabilidad del software en tiempo real (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                                                                                                                                                                 | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                                                                                                                                                         | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 5.1           | Asignar Tareas a los miembros del equipo de trabajo relacionadas con su rol, de acuerdo con el Plan de Proyecto actual.                                                                                                                                         | X                  |              |              |         |            |            |           | X         |
| 5.2           | Entender los Casos de Prueba y los Procedimientos de Prueba. Establecer o actualizar el entorno de pruebas.                                                                                                                                                     | X                  | X            | X            | X       | X          | X          | X         | X         |
| 5.3           | Integrar el Software usando Componentes de Software y actualizar los Casos de Prueba y Procedimientos de Prueba para las pruebas de integración, según sea necesario.                                                                                           | X                  | X            | X            | X       | X          | X          | X         | X         |
| 5.4           | Realizar pruebas de software utilizando Casos de Prueba y Procedimientos de Prueba para la integración y documentar los resultados en el Informe de Pruebas.                                                                                                    |                    |              |              | X       |            |            |           | X         |
| 5.5           | Corregir los defectos encontrados y realizar pruebas de regresión hasta que se alcancen los criterios de salida.                                                                                                                                                |                    | X            | X            | X       | X          | X          | X         |           |
| 5.6           | Actualizar el Registro de Trazabilidad, si es apropiado.                                                                                                                                                                                                        | X                  |              |              |         |            |            |           |           |
| 5.7           | Documentar la *Guía de Operación del Producto o actualizar la guía actual, si es apropiado. *(Opcional)                                                                                                                                                         | X                  |              |              |         |            |            |           |           |
| 5.8           | Verificar y obtener la aprobación de la *Guía de Operación del Producto, si es apropiado (ver SI.5.7)                                                                                                                                                           | X                  |              |              |         |            |            |           |           |
| 5.9           | Documentar la *Documentación del Usuario del Software o actualizar la actual, si es apropiado. *(Opcional)                                                                                                                                                      | X                  |              |              |         |            |            |           |           |
| 5.10          | Verificar y obtener la aprobación de la *Documentación del Usuario del Software, si es apropiado (ver SI.5.9)                                                                                                                                                   | X                  |              |              |         |            |            |           |           |
| 5.11          | Incorporar los Casos de Prueba y Procedimientos de Prueba, Software, Registro de Trazabilidad, Informe de Pruebas, *Guía de Operación del Producto y *Documentación del Usuario del Software a la Configuración del Software como parte de la base. *(Opcional) | X                  |              |              |         |            |            |           |           |

Tabla 11 – Tabla de Correspondencia de la actividad IS Integración y pruebas de software respecto a DevOps

Entrega del producto

La entrega del producto en el estándar ISO/IEC 29110 se centra en asegurar que el software cumpla con los criterios de aceptación establecidos y sea entregado adecuadamente al cliente, lo cual se complementa armónicamente con las actividades de DevOps, que buscan optimizar y automatizar los procesos de liberación y despliegue. En DevOps, la entrega continua permite que cada cambio validado a través de la construcción y pruebas sea rápidamente liberado y desplegado en entornos de producción, garantizando una operación eficiente y un monitoreo constante para identificar y resolver cualquier problema post-despliegue de manera inmediata. Esta integración resulta en un flujo de trabajo más ágil y eficiente, donde la entrega del producto no solo cumple con las expectativas del cliente, sino que también se adapta de manera continua a las mejoras y necesidades emergentes, asegurando un alto nivel de satisfacción y rendimiento del producto final (ISO/IEC, 2011).

| ISO/IEC 29110 |                                                                                                                         | Actividades DevOps |              |              |         |            |            |           |           |
|---------------|-------------------------------------------------------------------------------------------------------------------------|--------------------|--------------|--------------|---------|------------|------------|-----------|-----------|
| Tarea         | Descripción de la tarea                                                                                                 | Planificación      | Codificación | Construcción | Pruebas | Liberación | Despliegue | Operación | Monitoreo |
| 6.1           | Asignar Tareas a los miembros del equipo de trabajo relacionados con su rol, de acuerdo con el Plan de Proyecto actual. | X                  |              |              |         |            |            |           | X         |
| 6.2           | Entender la Configuración del Software.                                                                                 | X                  |              |              |         |            | X          | X         | X         |
| 6.3           | Documentar la Documentación de Mantenimiento o actualizar la actual.                                                    | X                  | X            | X            | X       | X          | X          | X         |           |
| 6.4           | Verificar y obtener la aprobación de la Documentación de Mantenimiento.                                                 | X                  |              |              |         |            |            |           |           |
| 6.5           | Incorporar la Documentación de Mantenimiento como base para la Configuración del Software.                              | X                  |              |              |         |            |            |           |           |
| 6.6           | Realizar la entrega de acuerdo con las Instrucciones de Entrega.                                                        |                    |              |              |         |            | X          | X         |           |

Tabla 12 – Tabla de Correspondencia de la actividad IS Entrega del producto respecto a DevOps

## 5. Resultados

La exploración de las prácticas de DevOps respaldada por la revisión de la literatura mostró una comprensión ampliada del proceso DevOps y su aplicación práctica. Se identificaron y documentaron las prácticas clave dentro de las actividades DevOps que promueven una integración y entrega continuas, así como la colaboración entre equipos de desarrollo y operaciones, destacando la compatibilidad con las tareas del perfil básico del estándar ISO/IEC 29110.

Mediante una matriz de correspondencia, se logró el análisis de cómo cada tarea del estándar ISO/IEC 29110 se relaciona con las actividades de DevOps. Este mapeo detallado mostró puntos en común, además de identificar oportunidades de mejora y adaptación de las tareas del estándar 29110 en el contexto de las prácticas de DevOps.

El estudio encontró que la implementación de DevOps dentro del marco de la ISO/IEC 29110 puede potenciar significativamente la eficiencia operativa, reducir el tiempo de comercialización y mejorar la respuesta a los problemas de una mejor forma. La evaluación de las prácticas de DevOps en el entorno del estándar 29110 sugiere que, mientras se mantiene la conformidad con el estándar, es posible adoptar un enfoque más flexible y adaptativo, que resulta en una mejora continua y en la entrega de un software que satisface las necesidades emergentes del mercado de una manera más eficaz.

Los hallazgos pueden justificar la adopción de DevOps como un complemento para mejorar la eficiencia, la calidad y la respuesta a los cambios en el entorno de desarrollo y operaciones de software, dando así un paso adelante en la evolución de las mejores prácticas de la ingeniería de software.

## 6. Conclusiones y trabajo futuro

La importancia de la presente investigación radica en el valor que tiene para las organizaciones comprender cómo las prácticas modernas de DevOps pueden ser incorporadas en marcos de trabajo estandarizados (ISO/IEC 29110 perfil básico). Al

realizar un análisis exhaustivo que relaciona directamente las actividades de DevOps con un estándar de calidad reconocido como el ISO/IEC 29110, la investigación busca proporcionar una guía práctica para la implementación efectiva de DevOps en organizaciones que desean adherirse a o ya siguen este estándar.

El presente estudio ha establecido que DevOps y el estándar ISO/IEC 29110 pueden coexistir de manera complementaria, ofreciendo un marco de trabajo robusto que aprovecha las fortalezas de ambos enfoques. Se ha identificado que, a pesar de sus diferencias, la agilidad y la automatización de DevOps se alinean con los procedimientos sistemáticos de la ISO/IEC 29110, contribuyendo a una mejor calidad y eficiencia en el desarrollo de software. Los hallazgos sugieren que la adopción de DevOps en organizaciones que siguen la ISO/IEC 29110 puede conducir a una mejora significativa en la entrega de software, con la adaptabilidad necesaria para responder a las demandas del mercado actual. La integración de DevOps con el estándar ISO/IEC 29110 ha demostrado ser especialmente beneficiosa para las entidades muy pequeñas, ya que facilita un enfoque escalable y flexible sin comprometer la calidad y la gobernanza del proyecto.

Como trabajo futuro, se está desarrollando una guía clara que ayude a la adopción de prácticas y facilite la implementación. Además, se considera relevante resaltar a partir de los resultados de este estudio se identifican como futuras líneas de investigación, estudios enfocados en estrategias específicas y estudios de caso detallados sobre la implementación de DevOps en organizaciones que están reguladas por el ISO/IEC 29110. Esto podría incluir el desarrollo de un conjunto de mejores prácticas y herramientas específicas para facilitar la transición y adaptación de los procesos ISO a entornos DevOps. Además, sería valioso explorar el impacto a largo plazo de esta integración en la calidad del software y en la satisfacción del cliente, así como en la capacidad de las organizaciones para mantenerse competitivas y ágiles en un panorama tecnológico en constante evolución. Finalmente, otra perspectiva podría realizar un análisis en el cual se contrasten no solo las sinergias sino las discrepancias entre ambos enfoques lo cual serviría de guía para las empresas para lograr complementar y potenciar la aplicación de ambos enfoques.

## Referencias

- Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the benefits of combining devops and agile.
- Bolívar, H., Valero, M.-F., Lopez, E., & Roza, R. (2023). Integration between HL7/CDA and ISO/IEC 29110 from a Systematic Review. 2023 Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI), 1-6. <https://doi.org/10.1109/CONIITI61170.2023.10324198>
- Bucena, I., & Kirikova, M. (2017). Simplifying the DevOps Adoption Process.
- Chen, L. (2017). Continuous Delivery: Overcoming adoption challenges. *Journal of Systems and Software*, 128, 72-86. <https://doi.org/10.1016/j.jss.2017.02.013>

- Dominguez-Acosta, M. F., & Garcia-Mireles, G. A. (2021). Identificación de Actividades Relacionadas con la Calidad de Software en Entornos de DevOps. 10th International Conference on Software Process Improvement, CIMPS 2021, 84-89. <https://investigadores.unison.mx/es/publications/identificaci%C3%B3n-de-actividades-relacionadas-con-la-calidad-de-soft>
- Erich, F., Amrit, C., Daneva, M. (2014). Report: DevOps Literature Review. University of Twente.
- Erich, F. M. A., Amrit, C., & Daneva, M. (2017). A qualitative study of DevOps usage in practice. *Journal of Software: Evolution and Process*, 29(6), Article 6. <https://doi.org/10.1002/smr.1885>
- França, B. B. N., Jeronimo, H., & Travassos, G. H. (2016). Characterizing DevOps by hearing multiple voices. In *Proceedings of the XXX Brazilian Symposium on Software Engineering* (pp. 53-62).
- Galvan-Cruz, S., Mora, M., Laporte, C. Y., & Duran-Limon, H. (2021). Reconciliation of scrum and the project management process of the ISO/IEC 29110 standard-Entry profile—An experimental evaluation through usability measures. *Software Quality Journal*, 29(2), 239-273. <https://doi.org/10.1007/s11219-021-09552-3>
- Ghantous, G. B., & Gill, A. (2017). DevOps: Concepts, Practices, Tools, Benefits and Challenges.
- Ghofrani, J., & Bozorgmehr, A. (2019). Migration to Microservices: Barriers and Solutions. En H. Florez, M. Leon, J. M. Diaz-Nafria, & S. Belli (Eds.), *Applied Informatics* (Vol. 1051, pp. 269-281). Springer International Publishing. [https://doi.org/10.1007/978-3-030-32475-9\\_20](https://doi.org/10.1007/978-3-030-32475-9_20)
- Guerrero, J., Zúñiga, K., Certuche, C., & Pardo, C. (2020). A systematic mapping study about DevOps. *Journal de Ciencia e Ingeniería*, 12(1), Article 1. <https://doi.org/10.46571/JCI.2020.1.5>
- ISO Calidad 2000. (2012). Guía para definir Procesos. ISO Calidad 2000. <https://isocalidad2000.com/2012/12/28/guia-para-definir-procesos/#:~:text=Resumiendo%2C%20tenemos%20los%20siguientes%20conceptos,las%20necesidades%20de%20los%20clientes>
- ISO/IEC. (2011). ISO/IEC TR 29110-5-1-2 Management and engineering guide: Generic profile group: Basic profile.
- Jabbari, R., bin Ali, N., Petersen, K., & Tanveer, B. (2016). What is DevOps? A systematic mapping study on definitions and practices. *Proceedings of the Scientific Workshop XP2016*, 1-11.
- Kamuto, M. B., & Langerman, J. J. (2017). Factors inhibiting the adoption of DevOps in large organisations: South African context. 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 48-51. <https://doi.org/10.1109/RTEICT.2017.8256556>

- Khan, M. S., Khan, A. W., Khan, F., Khan, M. A., & Whangbo, T. K. (2022). Critical Challenges to Adopt DevOps Culture in Software Organizations: A Systematic Review. *IEEE Access*, 10, 14339-14349. <https://doi.org/10.1109/ACCESS.2022.3145970>
- Lawrence Pfleeger, S. (2002). *Ingeniería de software. Teoría y práctica*. Ed. Prentice Hall. <http://dSPACE.scz.ucb.edu.bo/dSPACE/bitstream/123456789/599/3/3464.pdf>
- Lwakatare, L. E., Kuvaja, P., Oivo, M., Mäkinen, S., & Lassenius, C. (2016). DevOps adoption benefits and challenges in practice: A case study. In *Product-Focused Software Process Improvement* (pp. 590-597). Springer, Cham.
- López Peña, M. A. (2021). *DevOps Aplicado a Sistemas IoT: Definición e Implementación de Procesos Continuos de Monitorización y Retroalimentación* (PhD Thesis, ETSI). <https://oa.upm.es/id/eprint/68043>
- Mousaei, M., Gandomani, T.J. (2020). DevOps Approach and Lean Thinking in Agile Software Development: Opportunities, Advantages, and Challenges. *Journal of Soft Engineering and Intelligent Systems*.
- Moreno-Campos, E., Sanchez-Gordón, ML., Colomo-Palacios, R., de Amescua Seco, A. (2014). Towards Measuring the Impact of the ISO/IEC 29110 Standard: A Systematic Review. In: Barafort, B., O'Connor, R.V., Poth, A., Messnarz, R. (eds) *Systems, Software and Services Process Improvement. EuroSPI 2014. Communications in Computer and Information Science*, vol 425. Springer. [https://doi.org/10.1007/978-3-662-43896-1\\_1](https://doi.org/10.1007/978-3-662-43896-1_1).
- Muñoz, M. (2022). Boosting the Competitiveness of Organizations With the Use of Software Engineering. En *Research Anthology on Agile Software, Software Development, and Testing* (pp. 1838-1856). IGI Global. <https://www.igi-global.com/chapter/boosting-the-competitiveness-of-organizations-with-the-use-of-software-engineering/294547>
- O'Connor, R.V., & Laporte, C.Y. (2017). The evolution of the ISO/IEC 29110 set of standards and guides. *International Journal of Information Technologies and Systems Approach*.
- Perera, P., Silva, R., & Perera, I. (2017). Improve software quality through practicing DevOps. 2018-January, 13-18. Scopus. <https://doi.org/10.1109/ICTER.2017.8257807>
- Redondo, A. M. F., & Cárdenas, F. de J. N. (2022). DevOps: Un vistazo rápido. *Ciencia Huasteca Boletín Científico de la Escuela Superior de Huejutla*, 10(19), 35-40.
- Sharma, S., & Coyne, B. (2015). *DevOps para Dummies®* 2o Edición Limitada IBM.
- Ståhl, D., & Bosch, J. (2017). Continuous practices and DevOps: Beyond the buzz, what does it all mean? In *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (pp. 424-429). IEEE.

- Solís, I. S., & Tinoco, E. E. C. (2015). Compendio de Ingenieria de Software. Iván Soria Solís. <https://books.google.com/books?hl=es&lr=&id=3dZ5CwAAQBAJ&oi=fnd&pg=PP1&dq=somerville+en+ingenieria+de+software&ots=jQm-sucOwn&sig=u8-iawl3jeW72gJml7XJ1daP1HE>
- Takeuchi, M., Kohtake, N., Shirasaka, S., Koishi, Y., & Shioya, K. (2014). Report on an assessment experience based on ISO/IEC 29110. *Journal of Software: Evolution and Process*, 26.

# Estudio Terciario sobre Adopción de DevOps

Brian Pando<sup>1</sup>, Abel Silva<sup>1</sup>, Abraham Dávila<sup>2</sup>

brian.pando@unas.edu.pe; abel.silva@unas.edu.pe; abraham.davila@pucp.edu.pe

<sup>1</sup> Universidad Nacional Agraria de la Selva, 10131 Tingo María, Perú.

<sup>2</sup> Pontificia Universidad Católica de Perú, 15088, Lima, Perú.

DOI: 10.17013/risti.53.23-36

**Resumen:** DevOps es la respuesta frente a las disfunciones de las áreas de desarrollo y operaciones de software y DevOps tiene como sus prioridades la colaboración, productividad y automatización. Desde su aparición, se han realizado diversas publicaciones secundarias y pocos estudios terciarios, por lo que resulta útil mantener actualizado un estudio terciario sobre adopción de DevOps. El objetivo de este estudio es identificar y organizar estudios secundarios sobre la adopción de DevOps. Para ello, se siguió los pasos establecidos para un mapeo sistemático de la literatura. En la consulta a las seis bases de datos consideradas, se obtuvieron 346 artículos y luego del proceso quedaron 102 estudios secundarios seleccionados. Del análisis realizado, se tiene que los temas de las mejores prácticas y recomendaciones para la adopción de DevOps son las más abordadas. De ellas, las prácticas más reportadas son las de integración continua y entrega continua, el monitoreo y la automatización de las pruebas. Finalmente, existe buena predisposición por parte de los equipos de desarrollo, operación y gestión de adoptar DevOps en las organizaciones.

**Palabras-clave:** DevOps, Estudio Terciario, Integración Continua, Desarrollo Continuo.

## *A Tertiary Study on the DevOps Adoption*

**Abstract:** DevOps is the answer to the dysfunctions in the areas of software development and operations, and DevOps has collaboration, productivity and automation as its priorities. Since its appearance, there have been several secondary publications and few tertiary studies, so it is useful to keep a tertiary study on DevOps adoption up to date. The objective of this study is to identify and organize secondary studies on DevOps adoption. To do this, a systematic mapping study was performed. The six databases were consulted and 346 articles were obtained, resulting in 102 secondary studies being selected after the process. The analysis indicates that the most frequently discussed topics are best practices and recommendations for DevOps adoption. The practices that have been most frequently reported are continuous integration and delivery, monitoring, and test automation. Finally, there is a good willingness on the part of the development, operations, and management teams to adopt DevOps in organizations.

**Keywords:** DevOps, Tertiary study, Continuous Integration, Continuous Development.

## 1. Introducción

DevOps representa una respuesta a la realidad del trabajo descoordinado de las áreas de desarrollo de software y operaciones, y ha permitido la articulación de un conjunto de prácticas en la industria de software de aquella época (Debois, 2008). En particular, la práctica de integración continua es considerada como un factor crítico para su adopción (Mullaguru, 2015) y tiene como uno de sus pilares a las pruebas automatizadas (Virmani, 2015).

El desarrollo de software, bajo el enfoque de DevOps se basa en cuatro dimensiones (Lwakatare et al., 2015): colaboración, automatización, medición y monitoreo. Además, extiende las prácticas de los métodos ágiles, distribuyéndolas en 3 fases (Ebert et al., 2016): fase de construcción, en el que también se involucra las prácticas de integración continua; fase de despliegue, en el que se involucra mucho a gestión de la configuración; y fase de operación, donde se involucran actividades de monitoreo y registros de bitácora (logs).

La aplicación de DevOps ha permitido que tanto los equipos de desarrollo y operación colaboren en el trabajo, considerando tanto los aspectos de software (Soni, 2015) y negocio (Stillwell & Coutinho, 2015); siendo un logro relevante, la reducción del tiempo de lanzamiento del producto (Ebert et al., 2016). Asimismo, a medida que DevOps se interioriza en el equipo de desarrollo, presenta desafíos que los equipos deben enfrentar en la práctica (Senapathi et al., 2018). A pesar de ello, sus beneficios y ventajas superan los desafíos, incluso los de la transformación digital (Al-Zahrani & Fakieh, 2020), por lo que se ha convertido en un referente en el desarrollo de software (Mishra & Otaiwi, 2020).

En relación a la investigación sobre DevOps, se han realizado múltiples estudios primarios (Erich et al., 2014), (Rütz, 2019), secundarios (Guerrero et al., 2020) e incluso terciarios sobre DevOps (Arvanitou et al., 2022). Entre estos últimos, se puede resaltar de uno que intentó encontrar una definición más precisa del término “DevOps” (Arvanitou et al., 2022), y otro sobre las prácticas de DevOps en las empresas de software (Strieker & Volchkov Supervisor Martin Svensson Karlskrona, 2022), entre otros. En particular, en nuestro trabajo presentado en (Pando et al., 2024) se responde sobre aspectos relacionados a la adopción de DevOps.

En este estudio terciario, extendemos nuestro artículo previo (Pando et al., 2024), actualizando las respuestas a las preguntas de investigación sobre la adopción de DevOps. Mantenemos el uso de la guía de (Petersen et al., 2015) como si fuera un mapeo sistemático de la literatura (MSL) y cuyo objeto de estudio son los estudios secundarios adicionales. El resto de este artículo está organizado de la siguiente manera: en la Sección 2, el trabajo realizado en DevOps se presenta como estudios terciarios; en la Sección 3 se presenta el protocolo de ejecución para facilitar la replicabilidad del estudio; en la Sección 4 se presentan los resultados y se responden las preguntas de investigación planteadas; y, en la Sección 5, se presentan las conclusiones.

## 2. Trabajos relacionados

En la literatura científica, se han publicado una gran diversidad de artículos sobre adopción de DevOps en distintos contextos. Así, por ejemplo, se tienen: (i) estudios primarios como experiencias de adopción, beneficios de DevOps, problemas y logros de las organizaciones (Ghantous & Gill, 2017), (Riungu-Kalliosaari et al., 2016); (ii) estudios secundarios variados como prácticas, beneficios, calidad, desafíos (Guerrero et al., 2020), (Senapathi et al., 2018), (Mishra & Otaiwi, 2020); y finalmente, en general, (iii) pocos estudios terciarios como aplicación e investigación (Arvanitou et al., 2022), dependabilidad en DevOps (Andrade et al., 2023) y adopción de DevOps (Pando et al., 2024).

## 3. Diseño del estudio terciario

Este estudio, siguiendo el trabajo previo (Pando et al., 2024), se basa en el protocolo de (Petersen et al., 2015) estudio de mapeo sistemático (SMS) para ingeniería de software. Se mantuvo las preguntas de investigación que en el estudio previo. De la misma manera, se mantuvo la cadena de búsqueda en las seis bases de datos: Scopus, IEEE, ACM, Science Direct, Web of Science/Clarivate y Wiley. El diseño del estudio terciario se presenta a continuación.

### 3.1. Preguntas de investigación

Las preguntas de investigación, se mantuvieron del estudio previo (Pando et al., 2024) y que elaboró tomando como referencia (Arvanitou et al., 2022), (Neumann, 2022), (Pando & Dávila, 2022). Dichas preguntas de investigación son:

- RQ1. ¿Cuál es la evolución de los estudios secundarios sobre DevOps?
- RQ2. ¿Cuáles fueron los principales temas abordados en la adopción de DevOps?
- RQ3. ¿Cuáles son las prácticas utilizadas para adoptar DevOps?
- RQ4. ¿Cuáles son los marcos o enfoques utilizados al adoptar DevOps?
- RQ5. ¿Cuáles son las herramientas utilizadas en la adopción de DevOps?
- RQ6. ¿Cuál es el interés de los directivos de la empresa y de los equipos de desarrollo y operación en adoptar DevOps?
- RQ7. ¿Cuáles son los métodos ágiles que se utilizan al adoptar DevOps?
- RQ8. ¿Se menciona que DevOps ofrece una ventaja competitiva?

Para la categorización de los tipos de investigación en RQ1, utilizamos MSL de mapeo sistemático de la literatura y RSL de revisión sistemática de la literatura. Lo mismo se hizo con otras preguntas de investigación que requieren un categorizador.

### 3.2. Cadena de búsqueda

La cadena de búsqueda se creó utilizando la estrategia PI (Petersen et al., 2015) y tal como se presentó en (Pando et al., 2024). Las palabras clave para la cadena se eligieron a partir de una revisión inicial de los antecedentes del tema. La cadena de búsqueda fue:

*DevOps AND (“review of studies” OR “structured review” OR “systematic review” OR “literature review” OR “literature analysis” OR “in-depth survey” OR “literature*

*survey” OR “meta-analysis” OR “past studies” OR “subject matter expert” OR “analysis of research” OR “empirical body of knowledge” OR “systematic mapping” OR “overview of existing research” OR “body of published research” OR “study aggregation” OR “study synthesis” OR “critical review” OR “mapping study” OR “mapping literature”).*

Esta cadena tuvo que adaptarse a la sintaxis y restricciones de cada base de datos.

### 3.3. Criterios y proceso de selección

Para seleccionar los artículos secundarios se establecieron los siguientes criterios de inclusión y exclusión.

- CI1. El título y resumen del artículo están relacionados con el tema.
- CI2. El estudio está escrito en inglés.
- CI3. El estudio es una investigación secundaria de la literatura.
- CE1. El artículo está duplicado.
- CE2. El contenido completo del artículo no está disponible.

Para este estudio, se estableció utilizar los siguientes criterios de calidad (CC) (Kitchenham et al., 2010):

- CC1. ¿Están descritos y son apropiados los criterios de inclusión y exclusión de la revisión?
- CC2. ¿Es probable que la búsqueda bibliográfica haya cubierto todos los estudios relevantes?
- CC3. ¿Evaluaron los revisores la calidad/validez de los estudios incluidos?
- CC4. ¿Se describieron adecuadamente los datos/estudios básicos?

El proceso de selección se definió en tres etapas. La primera etapa fue la extracción de metadatos de bases de datos digitales. La segunda etapa fue la revisión de resúmenes, donde se aplicaron CI1, CI2, CI3 y CE1. La tercera etapa fue la revisión rápida de contenidos, donde se aplicaron CI2, CI3 y CE2. Finalmente, se aplicó la evaluación de la calidad (ver Apéndice A).

### 3.4. Amenazas a la validez

El análisis de las amenazas a la validez se realiza tomando como base los lineamientos de (Ampatzoglou et al., 2019) que consideran la validez de la selección del estudio, los datos y la investigación. Además, se mantiene lo señalado en (Pando et al., 2024).

Validez de la selección del estudio. Se mantuvo las preguntas de investigación planteadas, las mismas que se obtuvieron revisando un grupo de 15 estudios secundarios y terciarios relevantes. Asimismo, se mantuvo las palabras claves y los criterios de inclusión y exclusión establecidos en su oportunidad.

Validez de los datos. Se mantuvo las seis bases de datos relevantes para la ingeniería de software y se mantiene el hecho que pueden existir otras fuentes con estudios relevantes para este estudio terciario. El esquema de categorización basado en las preguntas de investigación se mantuvo durante esta segunda iteración del estudio terciario. Durante la categorización, uno de los autores revisó aleatoriamente para confirmar la categorización.

Validez de la investigación. Los tres autores están familiarizados con los temas de investigación en ingeniería de software y dos autores tienen experiencia en estudios secundarios. El estudio realizado es replicable considerando que se reportan todos los pasos seguidos para llegar a los resultados en el artículo previo (Pando et al., 2024). Además, la lista de estudios secundarios seleccionados se recoge en el Apéndice B. Los resultados pueden generalizarse, considerando lo indicado en el ítem anterior, ya que los estudios secundarios se cubren sin distinguir zona geográfica ni espectro temporal; esto último es considerando que los nuevos estudios se han adicionado al estudio anterior.

## 4. Resultados

El proceso de selección se realizó en base a la cadena de búsqueda aplicada a las bases de datos consideradas.

En la primera etapa, acumulando los resultados anteriores (Pando et al., 2024) y actuales, hasta el 28 de Feb de 2024, se identificaron 346 artículos: ACM (18), WoS (62), IEEE (38), ScienceDirect (21), Scopus (197) y Wiley (10). Luego de las etapas 2 y 3, se seleccionaron 102 estudios como estudios secundarios seleccionados (SSS). Para realizar la selección se utilizó la plataforma Parsifal (Freitas, 2021). La lista de los estudios secundarios seleccionados (ESS) se pueden ver en el Apéndice B. Las respuestas a las preguntas de investigación se presentan a continuación.

### 4.1. RQ1. ¿Cuál es la evolución de los estudios secundarios sobre DevOps?

Los estudios secundarios seleccionados, como se presenta en la Figura 1.a, se publican desde el 2014 y han mantenido un crecimiento continuo hasta llegar a un primer pico en el 2020 con 22 estudios secundarios, y un segundo pico el 2022 con 23 estudios secundarios. Esta situación evidencia que es un dominio en donde los investigadores han estado muy activos en el último quinquenio del 2019-2023 que tiene una media de 16 artículos por año, mientras que en 2014-2018 (quinquenio anterior) 3.8 artículos por año. Sin embargo, se nota una caída entre 2022 y 2023; y no se puede concluir aún para el 2024.

Adicionalmente, en lo que respecta a las publicaciones año por año de nuestro estudio, se nota que tiene el mismo patrón que el de prácticas ágiles presentado en (Neumann, 2022), lo cual tiene cierto sentido pues DevOps también es del mundo ágil.

Además, algunos detalles interesantes como en la Figura 1.b donde se observa que para el 2023, el número de Mapeos (9) es mayor que el de Revisiones (6), algo que se mantiene a la fecha en el 2024. De la Figura 1.c se obtiene que el 73% de publicaciones se concentran en Europa (44.1%) y Asia (29.4%). Finalmente, de la Figura 1.d se observa que el 67.4% son Revisiones y el resto (32.6%) Mapeos.

### 4.2. RQ2. ¿Cuáles fueron los principales temas abordados en la adopción de DevOps?

Al momento de consolidar los resultados, se obtuvo que los temas principales son: (i) implementación de DevOps (53), automatización (12), desafíos (12), factores críticos de éxito (8) y DevSecOps (8); los que en conjunto representan 91.1% (93/102) de los ESS. En

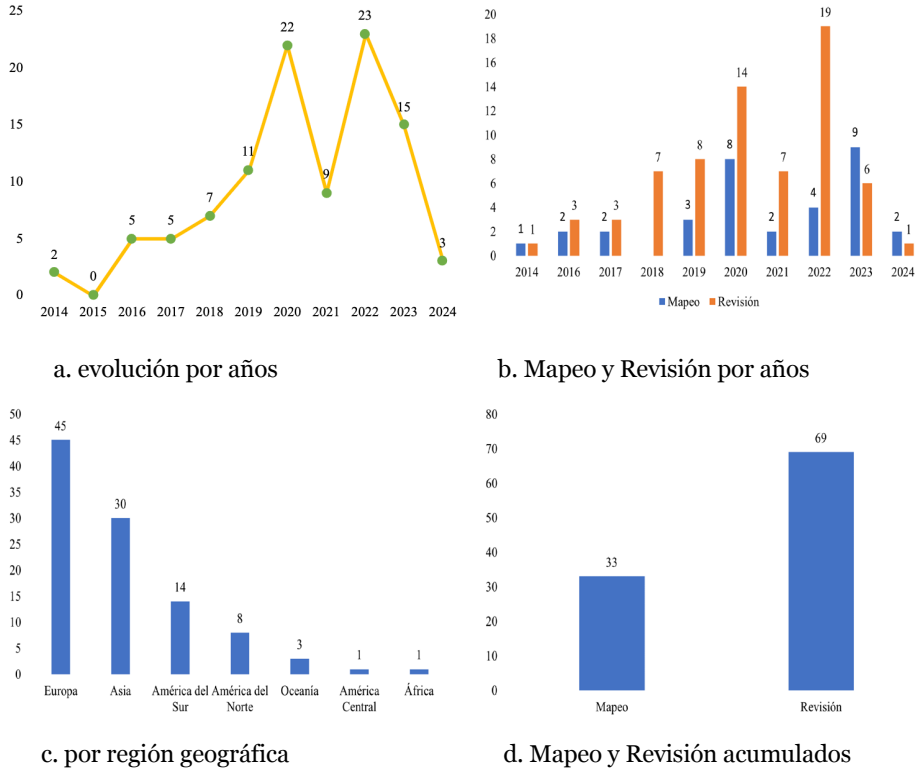


Figura 1 –Evolución de estudios secundarios seleccionados

el estudio previo (Pando et al., 2024), se habían identificado, además otros temas menos relevantes como: capacidades y prácticas, requisitos y colaboración, microservicios, calidad y despliegue, aprendizaje de maquina e internet de las cosas, entre otros.

Se observa que, en la Figura 2b, 53 ESS se centraron en la implementación de DevOps, es decir, en encontrar mejores prácticas y recomendaciones para lograr una implementación exitosa de DevOps. Los otros dos temas de interés son la automatización y lo desafíos en DevOps; que en el caso de la automatización es uno de los pilares que favorecer las prácticas de integración y despliegue continuo.

Además, reforzando lo identificado en el estudio previo (Pando et al., 2024), se tiene que existen pocos estudios secundarios que se enfoquen en buscar interacción con otras tecnologías o temas como IoT, seguridad o calidad, entre otros. Esto representa una oportunidad para incrementar la investigación sobre esos temas. Asimismo, también conviene resaltar que es necesario hacer investigaciones en la industria de software orientadas a la adopción de DevOps en el contexto de pequeñas empresas.

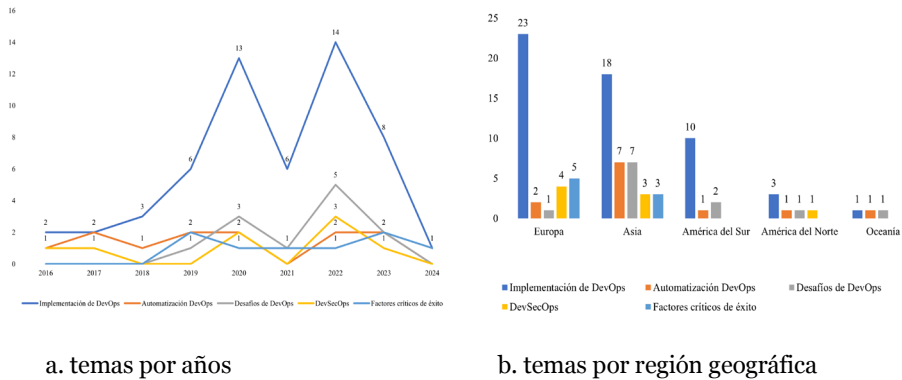


Figura 2 – Principales temas reportados en los ESS

### 4.3.RQ3. ¿Cuáles son las prácticas utilizadas para adoptar DevOps?

A partir de los ESS, se puede observar en la Figura 3, que las prácticas más estudiadas son las de integración continua (97), entrega continua (78), monitoreo continuo (76), automatización de pruebas (63), y despliegue continuo (58). Se debe tener claro que más de una práctica puede ser mencionada en los ESS, por lo que la suma es mayor que 102 ESS.

Para (Arvanitou et al., 2022), las prácticas más frecuentes son la implementación continua, las pruebas y el monitoreo de software; que son similares a las prácticas encontradas en este estudio. Además, (Arvanitou et al., 2022) destaca que el uso de estas prácticas es un requisito innegociable a la hora de adoptar DevOps. El estudio de (Neumann, 2022), reporta que las pruebas, la integración continua, las entregas frecuentes y la retroalimentación continua como las prácticas más utilizadas. También señala que estas prácticas están relacionadas con aspectos técnicos y de colaboración. Estos resultados son similares a los encontrados en este estudio.

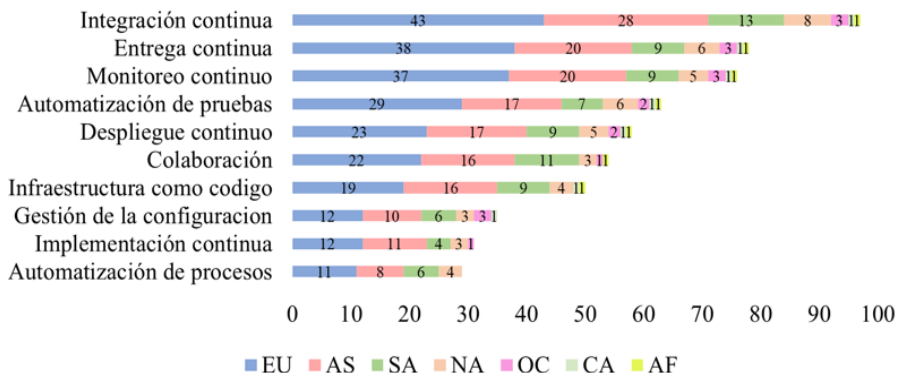


Figura 3 – Prácticas para DevOps según región geográfica y acumulado

#### 4.4.RQ4. ¿Cuáles son los marcos o enfoques utilizados al adoptar DevOps?

En el estudio previo (Pando et al., 2024), se presentaron varios marcos de trabajo, de los cuales, se han priorizado los más reportados. Entre ellos, según la Figura 4, se tiene a: el enfoque basado en la automatización (25), CAMS/CALMS (19), Modelo de Madurez DevOps (15), ITIL (12), SAFe (7), y el Fuzzy (5), los cuales representan el 81% (83/102) del total de ESS.

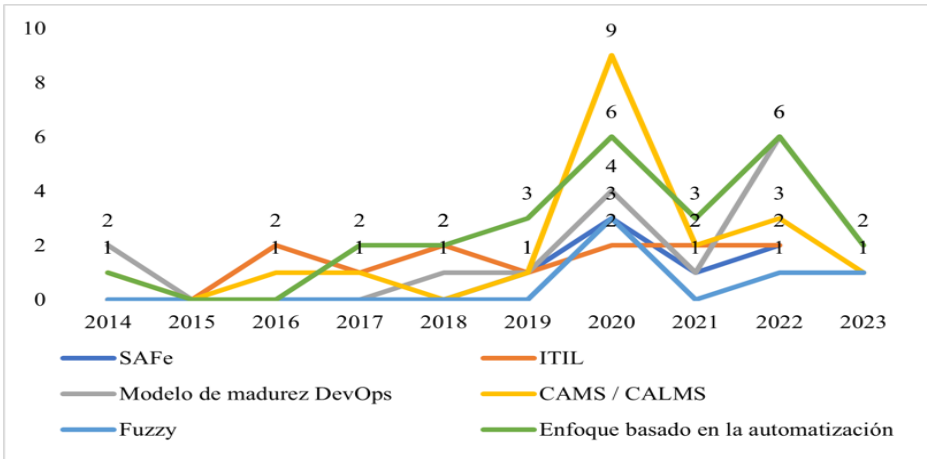


Figura 4 – Marcos de trabajo para adoptar DevOps

En la Figura 4, se puede observar el enfoque basado en automatización es uno de los marcos más reportados regularmente en el tiempo, teniendo un promedio de 4 artículos por año, en el último quinquenio. Mientras que CAMS/CALMS ha tenido un pico de 9 artículos, mientras que el promedio es de casi 1 artículo por mes. Por otro lado, el modelo de madurez de DevOps ha ido creciendo de manera sostenida, sin embargo, en el 2023 no hay resultados que lo refieran. En el caso de CALMS, que significa Cultura, Automatización, Liderazgo, Métricas y Compartir, aparece como un enfoque a la hora de adoptar DevOps. Este enfoque también menciona la automatización, pero agrega otros aspectos que son importantes para la adopción de DevOps.

En este estudio, los resultados muestran que existen enfoques y marcos conocidos en ingeniería de software que se están aplicando a DevOps. Sin embargo, aún no se ha definido un método para su adopción y desempeño en DevOps, lo cual concuerda con (Arvanitou et al., 2022), quien señala que no existe una metodología ampliamente establecida en DevOps.

#### 4.5.RQ5. ¿Cuáles son las herramientas utilizadas en la adopción de DevOps?

Según la Figura 5, existen varias herramientas que se han utilizado al adoptar DevOps. Debido a que DevOps es un enfoque basado en la automatización, en 83 estudios se menciona la herramienta Jenkins, que permite automatizar muchas tareas rutinarias. Un total de 69 estudios secundarios seleccionados mencionan a GitLab como herramienta

para el versionado del código fuente y alternativamente a Github con 31. En relación a la infraestructura, 78 estudios mencionan a Docker y 62 a Kubernetes. Además, para la gestión de infraestructura, Ansible se menciona en 73 estudios, Puppet en 53 estudios y Chef en 47 estudios.

Algunos estudios secundarios informan sobre otras herramientas como Jira, Azure DevOps o Nagios que respaldan la adopción de DevOps. Según (Guerrero et al., 2020), casi la mitad de los artículos encontrados mencionan que las herramientas tecnológicas son las que apoyan las prácticas a la hora de adoptar DevOps. Por otro lado, los resultados de este estudio, señalan que Jenkins, Docker, GitLab son algunas de las herramientas más reportadas que concuerdan con lo señalado por (Guerrero et al., 2020). Al igual que (Guerrero et al., 2020) y (Arvanitou et al., 2022), estas herramientas se encuentran principalmente en actividades continuas de integración e implementación, así como en monitoreo. Para (Arvanitou et al., 2022), la popularidad de estas herramientas podría generar oportunidades para desarrollar capacitación en habilidades para ingresar a la industria DevOps.

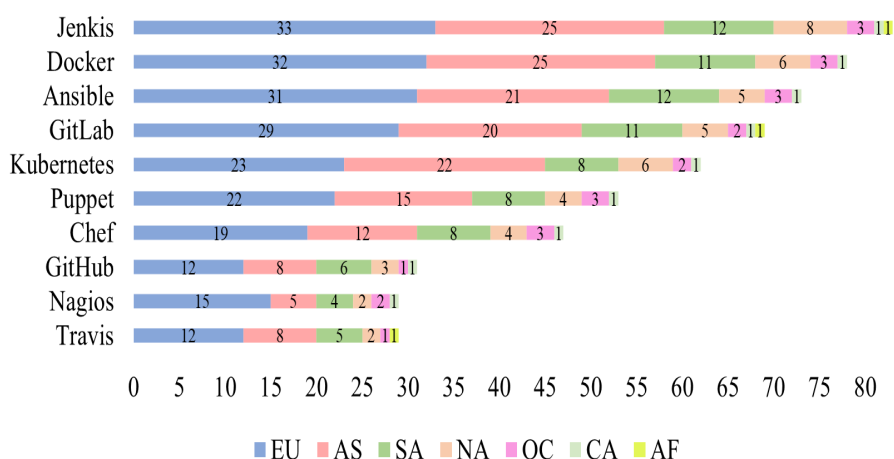


Figura 5 – Herramientas para DevOps según región geográfica y acumulado

#### 4.6.RQ6. ¿Cuál es el interés de los directivos de la empresa y de los equipos de desarrollo y operación en adoptar DevOps?

Se realizó un análisis narrativo, teniendo en cuenta las definiciones de (Cruzes et al., 2016), para sintetizar la relación con el interés de los directivos de la empresa, frases como “Las organizaciones eligieron adoptar DevOps”, “La organización abrazó gradualmente la adopción de DevOps” y “La adopción de DevOps en la organización fue exitoso”. Los resultados muestran que 98 (96%) estudios indican que las empresas están interesadas en adoptar DevOps. En el otro 4% no se pudo determinar si hay interés o no en adoptar DevOps.

En relación al interés de los equipos, se utilizaron frases con palabras como “interés”, “aceptación” o “aprobación” de los desarrolladores o equipos de desarrollo para adoptar DevOps. Los resultados muestran que se puede observar que 93 (91%) estudios indican que los equipos de desarrollo y operación están interesados en adoptar DevOps. En el otro 9% no se pudo determinar si hay interés o no en adoptar DevOps.

**4.7. RQ7. ¿Cuáles son los métodos ágiles que se utilizan al adoptar DevOps?**

Considerando que DevOps se basa en principios ágiles, resulta entendible que varios métodos ágiles se puedan adaptar a ellas. En particular en los ESS se reportan Scrum (66), Kanban (61), XP (43), Lean (34), DSSM (5), Crystal (2), con 1 DSSM, AUP y FDD; y no precisa 36 estudios. Estos resultados muestran que Scrum y Kanban son los más reportados, y que adoptar DevOps no sería un desafío cuando los equipos ya trabajan con métodos y técnicas que se basan en principios ágiles. En la Figura 6, se aprecia que las cuatro metodologías presentan un comportamiento similar, en particular Scrum y Kanban reportan las mismas cantidades de publicaciones hasta antes del 2021, por lo que en la Figura 6, da la impresión de ser sólo una.

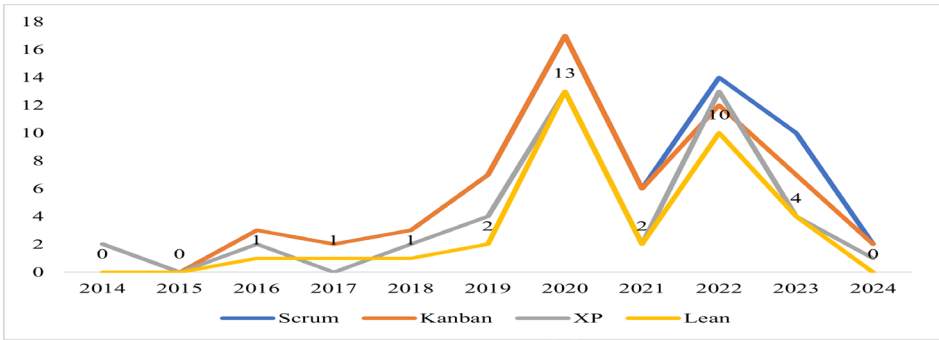


Figura 6 – Metodologías ágiles en la adopción de DevOps

Para (Hoda et al., 2017), Scrum es uno de los métodos más reportados en los artículos revisados en su estudio secundario, lo que coincide con los resultados de este estudio. Además, en este estudio, Kanban, XP y Lean se agregan a la lista de métodos utilizados cuando se trata de adoptar DevOps.

**4.8. RQ8. ¿Se menciona DevOps como una ventaja competitiva?**

Para responder a esta pregunta se busco frases como “el software hecho con DevOps, adquiere superioridad en el mercado”, “las empresas que adoptan DevOps reciben méritos y están mejor posicionadas en el mercado”. Los resultados revelan que 97 (95%) estudios mencionan a DevOps como una ventaja en la organización, pues son múltiples los beneficios que se adquieren cuando se implementa DevOps. El otro 5% de los artículos revisados no menciona si la adopción de DevOps le da a una organización una ventaja competitiva o no.

De los resultados de los estudios secundarios se puede establecer que existe una ventaja competitiva. Sin embargo, por la naturaleza de los estudios secundarios, es recomendable diseñar estudios, como encuestas, que establezcan tales ventajas desde la perspectiva de la gestión.

## 5. Conclusiones

En este artículo se presenta la extensión de un estudio terciario de la literatura sobre adopción de DevOps. El estudio se basa en la guía sugerida por (Petersen et al., 2015) teniendo como objetos de análisis a estudios secundarios. En el proceso de selección, se encontraron 346 estudios, de los cuales finalmente se seleccionaron 102 siguiendo el proceso establecido. A partir de dichos estudios secundarios se respondieron las ocho preguntas de investigación.

En el último quinquenio, la publicación de estudios secundarios se incrementó en gran medida, pasando de una media 3.8 artículos por año en el quinquenio 2014-2018 a una media de 16.0 artículos por año en el quinquenio 2019-2023. Lo que deja en evidencia el enorme interés que ha provocado hasta la actualidad esta temática.

Los estudios secundarios han presentado dificultades para encontrar mejores prácticas o recomendaciones para la adopción de DevOps. Entre las prácticas reportadas con frecuencia se encuentran la integración y entrega continua, el monitoreo continuo y la automatización de pruebas. Estos aspectos técnicos de DevOps se han estudiado más a fondo porque representan aspectos fundamentales de la parte técnica.

Para adoptar DevOps, los estudios revisados reportan un enfoque basado en la automatización, así como el marco CALMS y el modelo de madurez DevOps. Desde una perspectiva de proceso, la industria del software se beneficia de los dos últimos. CALMS refuerza la idea de que DevOps es una filosofía de trabajo que debe adoptarse teniendo en cuenta la cultura de los desarrolladores, y hablar de modelos de madurez es una señal de alineación con la estabilización de procesos y la mejora de la calidad asociada.

La mayoría de las herramientas reportadas son principalmente para infraestructura e integración y entrega continua. Los métodos ágiles reportados para la adopción de DevOps incluyen SCRUM, Kanban, XP y el enfoque Lean.

En cuanto a las personas, se establece que los equipos de desarrollo y operación tienen buena disposición para la adopción de DevOps en sus organizaciones. Respecto a las organizaciones, se establece que sí reconocen que adoptar DevOps representa una ventaja para la organización.

Como trabajo futuro, se ha previsto evaluar a un conjunto de empresas de software muy pequeñas, en la región Oriental del Perú, respecto a los hallazgos de este artículo e iniciar procesos de mejora de ser necesario para ese conjunto de organizaciones. Este esfuerzo servirá de marco para estudiar los distintos factores de adopción de DevOps en el conjunto descrito.

**Reconocimiento** Esta investigación fue realizada como parte de las actividades del Grupo de Investigación y Desarrollo en Ingeniería de Software (GIDIS) de la Pontificia

Universidad Católica del Perú y del grupo de Investigación de Ingeniería de Software (GINSOFT) de la Universidad Nacional Agraria de la Selva, como parte de la Red Peruana de Universidades (RPU) bajo el Project: gidis.RPU.005.PUCP.UNAS.2023.

**Apéndices:** <https://drive.google.com/drive/folders/1YhyGKGLxDDkTIlhpHz-6I6Ww62kU2c44>

## **Apéndice A: Resultados del proceso**

## **Apéndice B: Secondary studies selected**

## **Referencias**

- Al-Zahrani, S., & Fakieh, B. (2020). How DevOps Practices Support Digital Transformation. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(3), 2780–2788. <https://doi.org/10.30534/ijatcse/2020/46932020>
- Ampatzoglou, A., Bibi, S., Avgeriou, P., Verbeek, M., & Chatzigeorgiou, A. (2019). Identifying, Categorizing and Mitigating Threats to Validity in Software Engineering Secondary Studies. *Information and Software Technology*, 106, 201–230. <https://doi.org/10.1016/j.infsof.2018.10.006>
- Andrade, Á. J. D. C., Veloso, E., & Santos, G. (2023). What We Know About Software Dependability in DevOps - A Tertiary Study. *ACM International Conference Proceeding Series*, April, 178–187. <https://doi.org/10.1145/3629479.3629502>
- Arvanitou, E. M., Ampatzoglou, A., Bibi, S., Chatzigeorgiou, A., & Deligiannis, I. (2022). Applying and Researching DevOps: A Tertiary Study. *IEEE Access*, 10, 61585–61600. <https://doi.org/10.1109/ACCESS.2022.3171803>
- Cruzes, D. S., Moe, N. B., & Dyba, T. (2016). Communication between Developers and Testers in Distributed Continuous Agile Testing. *11th IEEE International Conference on Global Software Engineering, ICGSE 2016*, 59–68. <https://doi.org/10.1109/ICGSE.2016.27>
- Debois, P. (2008). Agile Infrastructure & Operations. *Agile 2008 Conference*. <http://www.jedi.be/presentations/agile-infrastructure-agile-2008.pdf>
- Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, 33(3), 94–100. <https://doi.org/10.1109/MS.2016.68>
- Erich, F., Amrit, C., & Daneva, M. (2014). Report: DevOps Literature Review. <https://doi.org/10.13140/2.1.5125.1201>
- Freitas, V. (2021). Parsifal. <https://parsif.al/>
- Ghantous, G. B., & Gill, A. Q. (2017). DevOps: Concepts, Practices, Tools, Benefits and Challenges. *Proceedings Of the 21st Pacific Asia Conference on Information Systems: Societal Transformation Through IS/IT, PACIS 2017*.
- Guerrero, J., Zúñiga, K., Certuche, C., & Pardo, C. (2020). A Systematic Mapping Study About DevOps. *Journal de Ciencia e Ingeniería*, 12(1), 48–62. <https://doi.org/10.46571/JCI.2020.1.5>

- Hoda, R., Salleh, N., Grundy, J., & Tee, H. M. (2017). Systematic Literature Reviews in Agile Software Development: A Tertiary Study. *Information and Software Technology*, 85, 60–70. <https://doi.org/10.1016/j.infsof.2017.01.007>
- Kitchenham, B., Pretorius, R., Budgen, D., Pearl Brereton, O., Turner, M., Niazi, M., & Linkman, S. (2010). Systematic Literature Reviews in Software Engineering – A Tertiary Study. *Information and Software Technology*, 52(8), 792–805. <https://doi.org/10.1016/j.infsof.2010.03.006>
- Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015). Dimensions of DevOps. In *International Conference on Agile Software Development* (pp. 212–217). Springer. [https://doi.org/10.1007/978-3-319-18612-2\\_19](https://doi.org/10.1007/978-3-319-18612-2_19)
- Mishra, A., & Otaifi, Z. (2020). DevOps and Software Quality: A Systematic Mapping. In *Computer Science Review* (Vol. 38). <https://doi.org/10.1016/j.cosrev.2020.100308>
- Mullaguru, S. N. (2015). Changing Scenario of Testing Paradigms using DevOps--A Comparative Study with Classical Models. *Global Journal of Computer Science and Technology*, 15(2), 23–27.
- Neumann, M. (2022). The Integrated List of Agile Practices - A Tertiary Study. In *Lecture Notes in Business Information Processing: Vol. 438 LNBIP* (pp. 19–37). Springer Science and Business Media Deutschland GmbH. [https://doi.org/10.1007/978-3-030-94238-0\\_2](https://doi.org/10.1007/978-3-030-94238-0_2)
- Pando, B., & Dávila, A. (2022). Software Testing in the DevOps Context: A Systematic Mapping Study. *Programming and Computer Software*, 48(8), 658–684. <https://doi.org/10.1134/S0361768822080175>
- Pando, B., Silva, A., & Dávila, A. (2024). DevOps Adoption: A Tertiary Study. In *Studies in Computational Intelligence* (Vol. 1135). [https://doi.org/10.1007/978-3-031-50590-4\\_3](https://doi.org/10.1007/978-3-031-50590-4_3)
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for Conducting Systematic Mapping Studies in Software Engineering: An Update. *Information and Software Technology*, 64, 1–18. <https://doi.org/10.1016/j.infsof.2015.03.007>
- Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps Adoption Benefits and Challenges in Practice: A Case Study. In P. Abrahamsson, A. Jedlitschka, A. N. Duc, M. Felderer, S. Amasaki, & T. Mikkonen (Eds.), *International Conference on Product-Focused Software Process Improvement: Vol. 10027 LNCS* (pp. 590–597). Springer, Cham. [https://doi.org/10.1007/978-3-319-49094-6\\_44](https://doi.org/10.1007/978-3-319-49094-6_44)
- Rütz, M. (2019). Devops: A Systematic Literature Review. *IT Management Seminar Paper Summer Term 2019*. <https://www.researchgate.net/publication/335243102>
- Senapathi, M., Buchan, J., & Osman, H. (2018). DevOps Capabilities, Practices, and Challenges. *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018, Part F1377*, 57–67. <https://doi.org/10.1145/3210459.3210465>

- Soni, M. (2015). End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery. 2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), 85–89. <https://doi.org/10.1109/CCEM.2015.29>
- Stillwell, M., & Coutinho, J. G. F. (2015). A DevOps Approach to Integration of Software Components in an EU Research Project. Proceedings of the 1st International Workshop on Quality-Aware DevOps, 1–6. <https://doi.org/10.1145/2804371.2804372>
- Strieker, J., & Volchkov, S. (2022). Broad DevOps Adoption and Organizational Performance of the Software Firm. Blekinge Institute of Technology. <http://urn.kb.se/resolve?urn=urn:nbn:se:bth-24285>
- Virmani, M. (2015). Understanding DevOps & bridging the gap from continuous integration to continuous delivery. 5th International Conference on Innovative Computing Technology, INTECH 2015, Intech, 78–82. <https://doi.org/10.1109/INTECH.2015.7173368>

# DevOps y la Medición de la Calidad del Producto de Software: Hallazgos Preliminares

Gabriel Alberto García-Mireles<sup>1</sup>, Nerina Peña Olivero<sup>2</sup>, Himer Avila-George<sup>2</sup>

[gabriel.garcia@unison.mx](mailto:gabriel.garcia@unison.mx); [nerinapolivero@gmail.com](mailto:nerinapolivero@gmail.com); [himer.avila@academicos.udg.mx](mailto:himer.avila@academicos.udg.mx)

<sup>1</sup> Departamento de Matemáticas, Universidad de Sonora, C. P. 83000, Hermosillo, México.

<sup>2</sup> Centro Universitario de los Valles, Universidad de Guadalajara, C. P. 46600, Guadalajara, México.

DOI: 10.17013/risti.53.37-52

**Resumen:** Las organizaciones de desarrollo de software están adoptando el enfoque DevOps como una estrategia para mantenerse competitivas. El impacto de estas iniciativas se relaciona con la reducción del tiempo de desarrollo mientras se mantiene la calidad del software. Sin embargo, pocos trabajos han señalado específicamente que aspectos de la calidad se han medido. En este trabajo, presentamos los resultados de un mapeo sistemático de literatura en donde identificamos las características de calidad del producto de software que se han estudiado en entornos DevOps. Los resultados se complementan con las iniciativas de medición del desempeño general de DevOps y aquellas que miden la calidad del producto de software. Dada la falta de trabajos en esta área, como primer paso en la definición de un proceso de medición de la calidad del producto en entornos DevOps, identificamos algunas métricas de mantenibilidad y factores contextuales por considerar en estas iniciativas.

**Palabras-clave:** DevOps, características de calidad de producto de software, medición, mantenibilidad.

## *DevOps and Software Product Quality Measurement: Preliminary Findings*

**Abstract:** Software development organizations are adopting the DevOps approach as a strategy to stay competitive. The impact of these initiatives is related to reducing development time while maintaining software quality. However, little research work has specifically indicated which aspects of quality have been measured. In this paper, we present the results of a systematic mapping study where we identify software product quality characteristics that have been studied in DevOps settings. The results are complemented by overall DevOps performance measurement initiatives and those that measure software product quality. Given the lack of research work in this area, as a first step in defining a product quality measurement process in DevOps settings, we identified some maintainability metrics and contextual factors to consider in these initiatives.

**Keywords:** DevOps, software product quality characteristics, measures, maintainability.

## 1. Introducción

El software impulsa la innovación y competitividad no solo de las industrias de las tecnologías de la información, sino de todos los sectores (Forsgren y Kernsten, 2018). Para muchas organizaciones, la adopción de enfoques ágiles y de DevOps se convierte en una estrategia para mantenerse competitivas (Forsgren y Kernsten, 2018). DevOps introduce un cambio organizacional en donde los grupos, tradicionalmente separados, de desarrollo y operaciones trabajan en entregas continuas del software (Debois, 2008; Ebert et al., 2016).

Los beneficios percibidos de la implementación de iniciativas basadas en DevOps reportan la reducción en el tiempo de entrega, menor número de defectos en los productos de software liberados, mejoras en el control de versiones, tiempo de comercialización más rápidos, mejoras en la productividad y mejora de la calidad del software, entre otras (Lazuardi et al., 2021). En una encuesta a desarrolladores de software, Offerman et al. (2022) reportan que los aspectos con mayor impacto positivo de la implementación de las prácticas de DevOps son la reducción del ciclo de desarrollo del producto, incremento en la colaboración, incremento en la efectividad del proceso de desarrollo y la detección temprana de defectos.

Aunque el impacto de las iniciativas DevOps se reporta positivo en diferentes indicadores organizacionales, el tema de la calidad del software se ha considerado desde la perspectiva del proceso (García-Mireles et al., 2015) dado que la calidad se mide en términos de la cantidad de defectos o fallas encontradas en las iniciativas DevOps que miden su impacto (Offerman et al., 2022; Visser y Hulstijn, 2023; Diamantopoulos, 2020; Bermon et al., 2022). Algunos estudios han señalado que la calidad del producto de software se ha estudiado poco y que existe la necesidad de abordarla en entornos DevOps (Domínguez-Acosta y García-Mireles, 2021; Mishra y Otaíwi, 2020). En particular, se recomienda estudiar el proceso de medición en las diferentes etapas de DevOps y cómo se podrían medir características de calidad como la usabilidad, eficiencia, mantenibilidad, portabilidad y seguridad (Mishra y Otaíwi, 2020).

Considerando lo expuesto anteriormente, el objetivo de este mapeo sistemático de literatura es conocer el estado actual de DevOps y la calidad del software, además identificar prácticas de DevOps informadas en la literatura que influyan específicamente en las características de calidad de producto que se encuentran representadas en el modelo de calidad de la norma ISO/IEC 25010 (ISO, 2011). Los resultados preliminares de este mapeo se reportaron en (Peña, Ávila & García-Mireles, 2023) en donde se describen las prácticas de DevOps que podrían estar correlacionadas con la calidad del producto de software y las características de calidad del software abordadas. Esta versión actualizada se centra en explorar cómo se realiza la medición en entornos DevOps y qué indicadores se recomiendan. Respecto de la calidad del producto de software se encontró poca evidencia de su medición. Así que, la investigación abordó la característica de mantenibilidad del producto de software para identificar cómo podría medirse en entornos DevOps. Sin embargo, la escasez de trabajos publicados en este tema llevó a recopilar un conjunto de métricas de mantenibilidad que se han aplicado en otros contextos como un primer paso para establecer un proceso de medición de la calidad del producto considerando el modelo ISO/IEC 25010 (ISO, 2011).

La estructura del documento es como sigue. En la sección 2 se presenta brevemente las principales revisiones de literatura que abordan la calidad del software en entornos DevOps. La sección 3 muestra los aspectos principales del proceso de estudio de mapeo sistemático realizado y cómo el estudio exploratorio permitió identificar algunas métricas de mantenibilidad que podrían ser un primer paso para establecer un proceso de medición de calidad del producto para dicha característica de calidad. En la sección 4 se mencionan los resultados principales del mapeo sistemático respecto de las características de calidad mencionadas en los estudios primarios, estándares utilizados y recomendaciones para abordar la calidad del producto. Además, se agregaron las sugerencias para medir el impacto de DevOps. En particular, la investigación se enfocó en identificar propuestas de la medición de la calidad del producto de software y cómo podría medirse la mantenibilidad. La discusión de los resultados se presenta en la sección 5. Finalmente, las conclusiones se presentan en la sección 6.

## 2. Antecedentes

En esta sección se presentan conceptos claves para la comprensión de la calidad del software y específicamente la calidad de producto de software, así como una visión general de algunos trabajos que abordan la relación de DevOps con la calidad del software.

La calidad del software se define como “el grado en el cual un producto de software satisface las necesidades enunciadas e implícitas cuando se usa en condiciones especificadas” (ISO, 2011). La calidad del software se puede estudiar desde distintas perspectivas y cada una de ellas enfoca su atención a diferentes aspectos que pueden ser medidos. Las dos perspectivas más utilizadas en la evaluación de la calidad del software son la basada en la vista de proceso y la vista de producto (ISO, 2011; García-Mireles et al., 2015).

La calidad de producto de software implica que hay un modelo de calidad subyacente que define un conjunto de características de producto (García-Mireles et al., 2015; ISO, 2011). La vista de producto de la calidad del software está basada en un modelo que define un conjunto de características de calidad de producto (Kitchenham & Pfleeger, 1996). En esta perspectiva, la calidad es vista como apropiada para su propósito y depende de las propiedades del producto que satisfacen las necesidades del usuario (Kitchenham & Pfleeger, 1996). La calidad de producto se mide considerando el contexto de la tarea o escenario (Kitchenham & Pfleeger, 1996). Para analizar y estudiar dichos escenarios, los modelos de calidad del software describen un conjunto de características de calidad que son útiles para identificar, medir y evaluar la calidad de producto de software (Mishra & Otaiwi, 2020; ISO, 2011).

Respecto de los modelos de calidad de producto de software, el ISO/IEC 25010 (ISO, 2011) proporciona una guía para identificar los requisitos de calidad de producto. Éste se compone por ocho características de calidad que debería tener un producto de software: adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad. Cada una de estas características contiene subcaracterísticas de calidad y, en conjunto, se tienen en cuenta al evaluar un producto de software.

Esta investigación aborda la característica de calidad denominada “mantenibilidad”, que se define como “el grado de efectividad y eficiencia con el cual los responsables del mantenimiento previsto pueden modificar un producto o sistema” (ISO, 2011). La mantenibilidad está compuesta por un conjunto de subcaracterísticas: modularidad, reusabilidad, analizabilidad, modificabilidad y comprobabilidad. La Tabla 1 presenta la traducción de las definiciones contenidas en el estándar y, entre paréntesis, se señala el nombre en inglés de cada una de las subcaracterísticas (ISO, 2011).

Las notas aclaratorias de dicha norma indican que las modificaciones pueden incluir correcciones, mejoras o adaptaciones del software a los cambios en el ambiente de operación. Además, dichas modificaciones incluyen aquellas que están bajo responsabilidad del personal de soporte especializado, personal de operaciones o usuarios finales. La mantenibilidad también incluye la instalación de actualizaciones y mejoras del software. Por lo tanto, es una característica clave de la calidad del producto que está estrechamente relacionada con el enfoque DevOps.

Por otra parte, el tema del impacto de DevOps en la calidad del producto de software se ha estudiado en distintos tipos de revisiones de literatura. Por ejemplo, en un estudio de mapeo sistemático Mishra y Otaïwi (2020) reportaron que la calidad del software tiene relación con aspectos de DevOps como la cultura, la colaboración, la automatización y el monitoreo. Además, una revisión exploratoria de literatura reporta diez prácticas que podrían influir la calidad del producto (Domínguez-Acosta y García-Mireles, 2021). Finalmente, Céspedes et al. (2020) señalan el interés por estudiar algunas características de calidad del producto, como confiabilidad y mantenibilidad. Además, las prácticas asociadas a DevOps, como el producto mínimo viable, la automatización de la implementación, la automatización de pruebas, cómputo en la nube y la cooperación en equipo, muestran una relación con la mejora en la calidad de producto de software. Sin embargo, ninguno de estos estudios muestra cómo se definen y miden las características de calidad del producto de software.

| Subcaracterística                  | Definición                                                                                                                                                                                                                                                                                             |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Modularidad<br>(Modularity)        | “Grado en el cual un sistema o programa de computadora se compone de componentes discretos tal que cambiar un componente tiene impacto mínimo en los restantes.” (ISO, 2011)                                                                                                                           |
| Reusabilidad<br>(Reusability)      | “Grado en el cual un activo puede ser usado en más de un sistema o en la construcción de otros activos.” (ISO, 2011)                                                                                                                                                                                   |
| Analizabilidad<br>(Analyzability)  | “Grado de efectividad y eficiencia con el cual es posible evaluar el impacto de un producto o sistema en un cambio previsto a una o más en sus partes, o diagnosticar un producto para determinar sus deficiencias o causas de fallas, o para identificar las partes por ser modificadas.” (ISO, 2011) |
| Modificabilidad<br>(Modifiability) | “Grado en el cual un producto o sistema puede ser efectiva y eficientemente modificado sin introducir defectos o degradar la calidad del producto existente” (ISO, 2011)                                                                                                                               |
| Comprobabilidad<br>(Testability)   | “Grado de efectividad y eficiencia con el que se pueden establecer criterios de prueba para un sistema, producto o componente y se pueden realizar pruebas para determinar si esos criterios se han cumplido.” (ISO, 2011)                                                                             |

Tabla 1 – Definición de las subcaracterísticas de mantenibilidad  
(tomado del ISO/IEC 25010 (ISO, 2011))

### 3. Metodología

Con base en lo expuesto anteriormente, el objetivo de este estudio de mapeo sistemático de literatura es identificar los estudios empíricos que muestren la relación entre las prácticas de DevOps y la calidad del producto de software. Se tomará como referencia de la calidad del producto las ocho características descritas en el modelo ISO/IEC 25010 (ISO, 2011). Puesto que los resultados de dicha revisión se presentaron en Peña, Ávila & García-Mireles (2023), este documento se centrará en la relación entre DevOps y la calidad del producto de software y se extenderá la revisión exploratoria de literatura a los enfoques de medición en DevOps y cómo se podría abordar la medición de la mantenibilidad del producto de software.

Los resultados del mapeo sistemático presentado en este artículo consideraron las guías propuestas por Kitchenham & Charters (2007). La descripción completa del proceso, incluyendo la lista completa de preguntas de investigación, criterios de selección de artículos primarios, procedimiento de extracción de datos se encuentra en (Peña, Ávila & García-Mireles, 2023).

En el mapeo sistemático, la pregunta principal de investigación es: ¿Qué impacto tienen las prácticas de DevOps en la calidad de producto de software de acuerdo con el modelo de calidad descrito por la norma ISO/IEC 25010? De esta pregunta general se derivan varias preguntas más específicas. En este artículo extendido, sólo presentaremos un resumen de los resultados obtenidos para contestar las siguientes:

- RQ-a. ¿Qué características de calidad de producto se informan en los estudios relacionados con DevOps?
- RQ-b. ¿Qué recomendaciones se sugieren para mejorar la calidad de producto al adoptar DevOps?

En el mapeo sólo consideramos la búsqueda automática en los siguientes recursos: ACM Digital Library, IEEE Xplore, Scopus y Web of Science. La cadena de búsqueda utilizada fue:

```
(DevOps) AND (9126 OR 25010 OR usability OR maintainability OR security OR efficiency OR portability OR reliability OR functionality OR compatibility OR "non-functional requirements" OR "nonfunctional requirements" OR nfr OR "quality characteristics" OR "quality objectives" OR "quality factors" OR "quality requirements" OR "quality attributes" OR "software quality" OR "product quality").
```

Con base en los resultados del mapeo sistemático, realizamos un análisis exploratorio del papel que desempeñan las métricas en iniciativas DevOps y en la calidad del producto de software. En particular, la investigación se centró en la mantenibilidad del producto de software. Para ello se realizó una consulta en Scopus con la cadena: (devops AND maintainability AND (measur\* OR metric\*)). Los resultados fueron verificados el 4 de abril de 2024 y Scopus devuelve sólo ocho registros. Uno de ellos, Tamburri et al. (2018), se enfoca en cómo se podría modificar el patrón Modelo-Vista-Controlador para medir sus eventos de forma automatizada. Sin embargo, nuestro interés es en evaluar el impacto de las prácticas de DevOps en la calidad del producto de software, por lo que este trabajo fue descartado.

Dada la escasez de estudios empíricos sobre este tema, la investigación se centró en métricas de la mantenibilidad del producto de software en cualquier contexto. Las mediciones encontradas se clasificaron con base en las subcaracterísticas de mantenibilidad del ISO/IEC 25010. Esta recopilación de métricas es el punto de referencia para elaborar un proceso de medición de la mantenibilidad del software que se abordará en la siguiente etapa de la investigación.

## 4. Resultados

La sección de resultados presenta los hallazgos principales del mapeo sistemático (Peña, Ávila & García-Mireles, 2023) relacionados con las características de calidad mencionadas en los estudios primarios y las recomendaciones para abordar la calidad del producto. Además, se presentan las principales iniciativas de medición en iniciativas DevOps. Por último, se presenta una lista inicial de métricas de la mantenibilidad que han sido utilizadas en distintos contextos como punto inicial para abordar un proceso de medición de la mantenibilidad del software.

### 4.1. Hallazgos principales del mapeo sistemático

Con base en los 19 estudios primarios seleccionados (Peña, Ávila & García-Mireles, 2023), la Figura 1 muestra las características de calidad de producto identificadas. Dos estudios primarios (Perera, Bandara, & Perera, 2016; Kato, Shimizu & Ishikawa, 2022) mencionan características de calidad como funcionalidad o adecuación funcional, fiabilidad, usabilidad, eficiencia de desempeño, mantenibilidad y portabilidad. Por su parte, hay referencias a la fiabilidad en (Bertolino et al., 2023) y eficiencia del desempeño en (Bezemer et al., 2019). Además, nueve estudios primarios abordan la seguridad del software con una tendencia hacia los estudios empíricos en el sector industrial (lista de referencias en Peña, Ávila & García-Mireles, 2023).

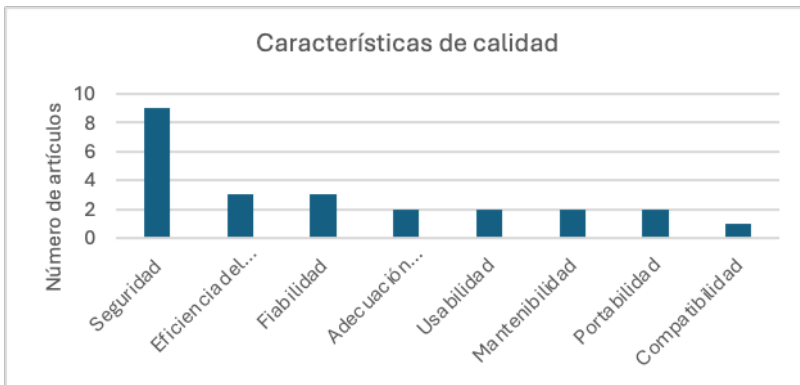


Figura 1 – Características de calidad mencionadas en los estudios primarios

Pocos estándares se citaron en los estudios primarios, entre ellos el IEC 62443 (Moyón et al., 2020) relacionado con la seguridad, el ISO/IEC 29110, mencionado por (Gunawan

& Budiardjo, 2021), y que está orientado a mejorar la calidad del software desde la perspectiva del proceso. Sin embargo, no se encontraron referencias a estándares como el ISO/IEC 25010 (ISO, 2011), cuyo modelo de calidad del producto puede usarse para definir los requisitos de calidad y evaluar el software en sus diferentes etapas de desarrollo.

Respecto de las recomendaciones para mejorar la calidad del producto de software, se sugiere implementar prácticas como el control de versiones y gestión de la configuración, pruebas automatizadas, integración continua automatizada y supervisión continua (Offerman et al., 2022; Kato, Shimizu & Ishikawa, 2022). Otros autores sugieren que todos los miembros del equipo sean conscientes de las responsabilidades asignadas a sus respectivos roles (Perera, Bandara, Perera, 2016). Además, se recomienda establecer los objetivos de calidad del producto con base en las características de calidad del software, y definir los logros de calidad en cada hito del proyecto (Kato, Shimizu & Ishikawa, 2022). Como resultado, se puede establecer un conjunto de pruebas apropiadas para los objetivos de calidad desde la planificación del proyecto y del aseguramiento de la calidad del software (Kato, Shimizu & Ishikawa, 2022).

Uno de los estudios primarios (Gunawan & Budiardjo, 2021) también proporciona recomendaciones para lograr mejoras en el producto, estas son: mapeo, diseño y separación de plataformas y partes de software para personalización mediante la utilización de mecanismos de línea de productos de software; aplicación de conceptos y modelos de diseño de software; y que se requieren pruebas como parte obligatoria en conjunto con el proceso de diseño de acuerdo con los requisitos del software. Por otra parte, las listas de entrada de productos deben incluir el registro de trazabilidad y el registro de riesgos como parte de los requisitos de cierres del proyecto realizados al comienzo del proyecto y actualizados continuamente a lo largo del proyecto.

El estudio de Bezemer et al. (2019), que aborda el rendimiento, recomienda asegurarse de que las herramientas se integren sin problemas en todos los procesos automatizados de DevOps, de manera que tanto el equipo de desarrollo y de operaciones puedan colaborar en la compilación e implementación de código en un entorno de producción. Dado que en su estudio observan que Jenkins CI es la herramienta de integración continua más popular recomiendan que los investigadores proporcionen complementos que permitan una fácil integración de sus herramientas de evaluación de rendimiento en Jenkins.

Algunos de los estudios que abordaron la seguridad también presentaron recomendaciones a tener en cuenta para lograr una mejora de la seguridad en DevOps. El trabajo de Tomas, Li & Huang (2019) recomienda que la seguridad se puede potenciar con un conjunto de herramientas adecuadas, que debe existir un estándar sobre cómo configurar pruebas de seguridad automatizadas, que podría ser beneficioso para las empresas obtener una descripción general de las características de seguridad que ofrecen los proveedores de servicios en la nube, pero también cuáles son las limitaciones y que lo más importante de la seguridad es que los desarrolladores estén capacitados en ella. Similarmente, (Rahman & Williams, 2016) recomiendan el monitoreo automatizado, las pruebas y la implementación automatizadas de software. Además, sugiere que actividades como la colaboración supervisada con el equipo de seguridad podría ayudar a que la implementación automatizada sea beneficiosa para la seguridad del sistema como una actividad de DevOps (Rahman & Williams, 2016).

Otras recomendaciones para mejorar la seguridad sugieren que el personal que diseñe los casos de prueba no sea el mismo programador del componente de software (Mohan, Othmane & Kres, 2018) ya que los casos de prueba funcionan como guardianes de los otros entornos. Además, recomiendan que las partes interesadas deben iniciar y aprobar una solicitud de cambio antes de la implementación en producción y que debe haber derechos de acceso separados para cada uno de los tres entornos: acceso de desarrollo, acceso de TIC y acceso de producción (Mohan, Othmane & Kres, 2018). Un desarrollador no deberá tener derecho de acceso a los entornos de prueba y producción. Finalmente, se propone agregar un rol de evaluador de seguridad independiente (Zeller, 2021).

#### 4.2. Iniciativas de medición en entornos DevOps

Aunque no hay una definición estándar de las etapas del proceso de DevOps (Cappizzi et al., 2019; Alnafessah et al., 2021), algunos autores las definen como la planeación, el desarrollo, la verificación, la prueba, la liberación, el despliegue, la operación y el monitoreo (Cappizzi et al., 2019; Bermon et al. 2022). De estas etapas, para medir la calidad del software podría considerarse el monitoreo. Esta etapa consiste en evaluar los datos relacionados con el despliegue continuo de las aplicaciones de software con la finalidad de encontrar fallas y proporcionar retroalimentación para mejorar el software y guiar su evolución (Alnafessah et al. 2021).

Para evaluar el desempeño de DevOps, las métricas más frecuentes abordan la frecuencia de despliegue, el cambio en el tiempo de espera, la restauración del servicio y el cambio en las tasas de falla (Bermon et al., 2022). Similarmente, para determinar el impacto de DevOps en la entrega del software, en los equipos y en la satisfacción del cliente, se requiere determinar el tiempo de espera, la frecuencia del despliegue, el tiempo promedio de restauración y el porcentaje de cambios fallidos (Offerman et al., 2022). Para Humble y Molesky (2011) DevOps se guía por cuatro conceptos rectores: la cultura, la automatización, la medición y el intercambio. En particular, la medición permite determinar que los objetivos de liberación y mejora se han alcanzado. El proceso de la medición incluye aspectos relacionados con el negocio, como rentabilidad, hasta aspectos a nivel de la implementación del software como número de transacciones por unidad de tiempo (Humble y Molesky, 2011). Este esfuerzo implica una selección cuidadosa de los indicadores de desempeño que fomenten la productividad del equipo de desarrollo y que eliminen cuellos de botella en el proceso de desarrollo (Humble y Molesky, 2011).

Wilkes et al. (2023) señalan que las mediciones del *DevOps Research and Assessment* (DORA) han sido ampliamente aceptadas por la industria del software y se utilizan para medir el desempeño de DevOps. El marco de medición DORA muestra cuatro métricas que se correlacionan con el desempeño de DevOps: frecuencia de despliegue, tiempo de espera para los cambios, tiempo promedio de recuperación, y tasa de fallos por modificaciones. Por su parte, Amaro et al. (2023) identifican y validan 24 mediciones que permiten determinar los resultados de adoptar iniciativas DevOps. Entre ellas, se encuentran mediciones relacionadas con la calidad del producto como tasa de incidentes y errores de producción, volumen de tiques del cliente, tiempo medio entre fallas y cobertura de pruebas automáticas de código. Sin embargo, se informa que las organizaciones enfrentan dificultades para rastrear las mediciones y se requiere mucho esfuerzo y experiencia para obtener estos datos (Amaro et al., 2023). Por lo

tanto, implementar un proceso de medición representa un reto para la mayoría de las organizaciones de software. Por otra parte, las organizaciones no sólo deben priorizar los indicadores de la calidad del producto, sino también las mejoras en la cultura y mejoras del proceso (Amaro et al. 2023).

Por su parte, Diamantopoulos et al. (2020) señalan que el monitoreo efectivo del proceso de desarrollo requiere analizar la calidad del software y que la evaluación se puede automatizar. Por ejemplo, al identificar las violaciones a las prácticas de codificación aceptadas. Además, Visser y Hulstijn (2023) sugieren una adaptación del modelo DMAIC (define, mide, analiza, mejora y controla) para establecer un proceso que permita el monitoreo del desempeño de equipos ágiles/DevOps para que el mejoramiento sea continuo. En particular, considera indicadores de la estabilidad del producto, como tasa de fallos por modificaciones, tiempo promedio de recuperación, y densidad de defectos. En conclusión, las iniciativas de medición de DevOps se han centrado en evaluar el impacto del proceso en diferentes indicadores de interés para la organización. Por su parte las fallas o defectos se utilizan en la perspectiva de procesos para medir la calidad (Kitchenham y Pfleeger, 1996; García-Mireles et al., 2015).

Otras iniciativas podrían responder a la necesidad de mejorar la calidad del producto de software. Algunas investigaciones abordan cómo integrar prácticas de seguridad en DevOps. Por ejemplo, DevSecOps es definido como la integración de las prácticas de seguridad en DevOps (Mohan y Sthmane, 2016). Prates et al. (2019) señalan que la implementación de DevOps podría comprometer la seguridad del software debido a ciclos de desarrollo más rápidos e incremento del número de despliegues. Por otra parte, para desarrollar tecnología de mejora de la seguridad, su revisión de literatura reporta nueve métricas que sirven para medir la efectividad de DevSecOps (Prates et al., 2019). Además, Mallouli et al., 2020 señalan que se requieren nuevas métricas de seguridad para monitorear el canal de DevSecOps para alertar y evitar posibles incidentes de seguridad. Sugieren un enfoque basado en aprendizaje de máquina para incluir controles de seguridad en las herramientas de soporte a las prácticas de DevSecOps (Mallouli et al., 2020).

Finalmente, Snyder y Curtis (2017) reportan la experiencia de implementar DevOps y de evaluar el impacto en la calidad estructural del software. La calidad estructural del software considera aspectos como robustez, seguridad, eficiencia del desempeño, facilidad de cambio, y transferibilidad que se agregaron en el Índice de Calidad Total. Se reporta que las mediciones de tamaño y calidad estructural del software contribuyeron a la productividad y a la reducción del tiempo de desarrollo. Sin embargo, no se indican las métricas específicas usadas para medir los aspectos de calidad señalados.

### 4.3. Medición de la mantenibilidad del software

Como se mencionó previamente en la sección de metodología, hay escasez de publicaciones científicas que aborden la relación entre prácticas de DevOps y mantenibilidad del software. Además, hay poca información respecto de cómo medir la mantenibilidad en este contexto. Por lo tanto, se realizó una búsqueda exploratoria para identificar métricas de mantenibilidad que se hayan utilizado en otros contextos.

En particular, se encontraron métricas que se han evaluado empíricamente en aplicaciones web, aplicaciones móviles, en software orientado a objetos, entre otros. Las métricas

encontradas se clasificaron tomando como base las subcaracterísticas de calidad de la mantenibilidad descritas en el ISO/IEC 25010 (ISO, 2011). Algunas métricas pueden ser consideradas en varias subcaracterísticas de la mantenibilidad. La Tabla 2, a manera de ejemplo, muestra algunas métricas de mantenibilidad que se han usado para evaluar aplicaciones web desde el diseño del software y la codificación, según las propuestas de Ghosheh, Black & Qaddour (2008) y Chae et al., (2007). Como se observa, la selección de las métricas está fuertemente influenciada por las características del software, la etapa del proceso de desarrollo y las metas de información que se persigan. Por lo tanto, un análisis apropiado de estas métricas debería considerar factores contextuales particulares y las necesidades de información de la organización.

Por otra parte, un conjunto de métricas ampliamente usado es la propuesta de Chidamber y Kemerer (1994) que incluye métricas para código orientado a objetos en donde se pueden usar las métricas como NOR (número de clases raíz), DIT (profundidad del árbol de herencia), y WMC (métodos ponderados por clase para medir la complejidad de los métodos).

| Subcaracterística | Métricas (Diseño)<br>Ghosheh, Black & Qaddour<br>(2008)                                                                                                                                 | Métricas (Código)<br>Chae et al. (2007)                                               |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| Analizabilidad    | Métricas de tamaño: número de páginas web, número de páginas del servidor, número de formularios<br>Métricas de complejidad: número total de enlaces, número total de relaciones submit | Tamaño: LOC (Líneas de código)<br>Acoplamiento: RFC (Medida de respuesta a una clase) |
| Comprobabilidad   | Métricas de complejidad                                                                                                                                                                 | Tamaño: LOC (Líneas de código)<br>Acoplamiento: RFC (Medida de respuesta a una clase) |
| Modificabilidad   | Métricas de tamaño<br>Métricas de complejidad                                                                                                                                           | Tamaño: LOC (Líneas de código)                                                        |
| Modularidad       | Acoplamiento de control<br>Acoplamiento de datos                                                                                                                                        | Acoplamiento: DAC –<br>Acoplamiento de abstracción de datos                           |
| Reusabilidad      | Número de relaciones de inclusión sobre el número de páginas web.                                                                                                                       | ---                                                                                   |

Tabla 2 – Algunas métricas para evaluar la mantenibilidad de aplicaciones web (basado en Ghosheh, Black & Qaddour (2008) y Chae et al. (2007))

Para implementar un marco de medición, se deberían tomar en cuenta los requisitos de las mediciones, como: que sean relevantes para el negocio y para el equipo DevOps, que sean medibles objetivamente, confiables y comparables (Visser y Hulstijn, 2023). Por otra parte, el proceso de medición debería estar bien integrado con el proceso de desarrollo existente (Visser y Hulstijn, 2023). Además, los factores que pudieran impactar al desempeño del equipo deberían poder identificarse (Visser y Hulstijn, 2023), por ejemplo, los señalados por Gaorusi et al. (2019). Entre más temprano más temprano una organización inicie su proceso de medición, más pronto podrá establecer

una línea base para medir las mejoras relativas de iniciativas como DevOps (Forsgren y Kernsten, 2018).

Realizar la evaluación a través de mediciones es un reto cuando se procura recolectar medidas de todo el canal de entrega del software debido al uso de múltiples sistemas que se han introducido al proceso de desarrollo para implementar prácticas de DevOps y que recolectan datos a través de las distintas áreas de la organización (Forsgren y Kernsten, 2018). Las decisiones del negocio deberían estar basados en una recolección de datos completa, correcta y exhaustiva que utilice modelos de análisis apropiados (Forsgren y Kernsten, 2018). La decisión sobre el conjunto de métricas por calcular es una tarea difícil que involucra considerar una serie de factores como las características de cada proyecto, el perfil de los desarrolladores y el personal de operaciones, y otros factores contextuales (Diamantopoulos et al., 2020). Además, se requiere un análisis periódico de cada métrica para verificar su confiabilidad con los cambios que se realizan durante el proyecto (Diamantopoulos et al., 2020).

## 5. Discusión

Respecto de los hallazgos del mapeo sistemático de literatura, los estudios primarios no mencionan la utilización de algún modelo de calidad de producto, tampoco definen las características, ni se tiene información en todos los casos de cómo fueron medidas. Solo el estudio (Bezemer et al., 2019), que aborda el rendimiento, lo define y según la definición que utilizan se podría traducir a eficiencia de desempeño. Por otra parte, la mayoría de los estudios primarios (9 de 19) abordan la seguridad, mientras que las otras características de calidad del software están abordadas hasta en un máximo de tres estudios primarios. Por lo tanto, analizar el impacto de DevOps en las características de calidad del producto es un tema abierto a la investigación.

Por su parte, las recomendaciones que se encontraron en los estudios primarios señalan que las pruebas automatizadas, implementaciones continuas y automatizadas, la colaboración y comunicación entre los miembros del equipo son las prácticas más sugeridas y que, a opinión de los estudios empíricos seleccionados, tienen mayor impacto positivo en la calidad de un producto de software. Estas recomendaciones coinciden, además, con ser prácticas de DevOps de las más mencionadas como potenciales para alcanzar mejoras en la calidad de producto de software (Mishra & Otaiwi, 2020; Céspedes et al., 2020).

Respecto de las iniciativas de medición en entornos DevOps se identifican dos áreas de interés relevantes. Por una parte, la necesidad de demostrar que las iniciativas de DevOps son efectivas ha llevado a la implementación de marcos de medición que utilizan indicadores como el tiempo de espera, la frecuencia de despliegue, el tiempo promedio de restauración del servicio y el porcentaje de cambio fallidos (Offerman et al. 2022). Por otra parte, la atención a las amenazas a la seguridad ha llevado al desarrollo de DevSecOps (Mohan y Sthmane, 2016), en donde las iniciativas tratan de establecer controles de seguridad en el código y establecer nuevas métricas apropiadas para este entorno (Prates et al., 2019; Mallouli et al., 2020). Sin embargo, estas iniciativas están orientadas a la perspectiva de proceso para medir al software. Pocos esfuerzos se han realizado para medir la calidad del producto (Snyder y Curtis, 2017).

Respecto de la mantenibilidad del software en DevOps, la revisión exploratoria de literatura señala que hay muy pocas publicaciones que aborden este tema. Como se informa en el ISO/IEC 25010 (ISO, 2011) la mantenibilidad del software está estrechamente relacionada con varias prácticas de DevOps. Por lo tanto, es un área de oportunidad para investigar en qué medida DevOps contribuye a la mantenibilidad del software y establecer la relación entre reducir los tiempos de liberación del producto con su mantenibilidad. Por otra parte, las métricas de mantenibilidad reportadas en la Tabla 2 han sido validadas en contextos distintos al entorno DevOps. Como se señaló previamente, éstas necesitan evaluarse para ver si son apropiadas para los ambientes DevOps y validarse en empresas que hayan adoptado este tipo de enfoque de desarrollo.

Las limitaciones principales de este estudio se deben a que la revisión sólo utilizó la búsqueda automática en cuatro bases de datos que son comunes en la investigación en ingeniería de software. Por lo tanto, no se puede establecer que se realizó una búsqueda exhaustiva. Por otra parte, las búsquedas en Scopus de las iniciativas de medición en DevOps y la mantenibilidad son de naturaleza exploratoria. Por lo tanto, tampoco se puede establecer la completitud del proceso de búsqueda. Sin embargo, lo que si se muestra con estos resultados es un área de oportunidad para abordar la calidad del producto en entornos DevOps, dado el resultado de las revisiones sistemáticas analizadas que son consistentes con los resultados presentados en este trabajo.

## 6. Conclusiones

Respecto del mapeo sistemático, el análisis preliminar mostró pocos estudios empíricos, en entornos DevOps, que abordaran la calidad del producto del software en términos de las características descritas en el ISO/IEC 25010 (ISO, 2011). Pocos estudios definen explícitamente las características de calidad o citan a un estándar apropiado. Además, no se encontraron especificaciones operativas de las métricas usadas para determinar la relación entre las prácticas de DevOps y la calidad del software. Respecto de las iniciativas de medición en DevOps, éstas tienden a medir la calidad del producto desde la perspectiva del proceso. Hay pocos trabajos publicados que indiquen cómo medir la mantenibilidad en entornos DevOps.

Aunque la seguridad es la característica de calidad que se encontró con mayor frecuencia en los estudios primarios, aún falta más trabajo para definir métricas apropiadas y herramientas para usarse en entornos DevOps. Además, el resto de las características de calidad del ISO/IEC 25010, que están definidas por consenso, podrían fundamentar otras investigaciones sobre la calidad del producto de software en dichos entornos. Respecto de la medición de la calidad del producto, la revisión exploratoria de literatura muestra que hay oportunidades para desarrollar procesos de medición.

## Referencias

- Alnafessah, A., Gias, A.U., Wang, R., Zhu, L., Casale, G., & Filieri, A. (2021). Quality aware DevOps research: Where do we stand? *IEEE Access*, 9, 44476-44489. <https://doi.org/10.1109/ACCESS.2021.3064867>

- Amaro, R., Pereira, R., & da Silva, M. M. (2023). Capabilities and metrics in DevOps: A design science study. *Information & Management*, 60(5), 103809.
- Bermon Angarita, L., Fernández Del Carpio, A. & Osorio Londoño, A. A. (2022). A Bibliometric Analysis of DevOps Metrics. *DESIDOC Journal of Library & Information Technology*, 42(6). pp. 387-396.
- Bertolino, A., Angelis, G. D., Guerriero, A., Miranda, B., Pietrantuono, R., & Russo, S. (2023). DevOpRET: Continuous reliability testing in DevOps. *Journal of Software: Evolution and Process*, 35(3), e2298.
- Bezemer, C. P., Eismann, S., Ferme, V., Grohmann, J., Heinrich, R., Jamshidi, P., Shang, W., van Hoorn, A., Villavicencia, M., Walter, J. & Willnecker, F. (2019). How is performance addressed in DevOps?. In *Proceedings of the 2019 ACM/SPEC International Conference on Performance Engineering* (pp. 45-50).
- Capizzi, A., Distefano, S. & Mazzara, M. (2019). From devops to devdataops: Data management in devops processes. In *International Workshop on Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment*, pp. 52-62. [https://doi.org/10.1007/978-3-030-39306-9\\_4](https://doi.org/10.1007/978-3-030-39306-9_4)
- Céspedes, D., Angeleri, P., Melendez, K. & Dávila, A. (2020). Software Product Quality in DevOps Contexts: A Systematic Literature Review, In: Mejia, J., Muñoz, M., Rocha, Á. A., Calvo-Manzano, J. (eds) *Trends and Applications in Software Engineering. CIMPS 2019. Advances in Intelligent Systems and Computing*, vol 1071. Springer, Cham. <https://doi.org/10.1007/978-3-030-33547-2>, 2020.
- Chae, H. S., Kim, T. Y., Jung, W. S., & Lee, J. S. (2007). Using metrics for estimating maintainability of web applications: An empirical study. In *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)* (pp. 1053-1059). IEEE.
- Chidamber, S.R., Kemerer, C.F. (1994)- A metrics suite for object oriented design. *IEEE Trans. Softw. Eng.* 20(6), 476-493.
- Debois, P. (2008). Agile infrastructure and operations: how infra-gile are you?. In *Agile 2008 Conference*. pp. 202-207.
- Diamantopoulos, T., Papamichail, M. D., Karanikiotis, T., Chatzidimitriou, K. C., & Symeonidis, A. L. (2020). Employing contribution and quality metrics for quantifying the software development process. In *Proceedings of the 17th International Conference on Mining Software Repositories*. pp. 558-562.
- Domínguez-Acosta, M. F. & García-Mireles, G. A. (2021). Identifying Activities for Enhancing Software Quality in DevOps Settings. *10th International Conference On Software Process Improvement (CIMPS)*, Torreón, Coahuila, Mexico, 2021, pp. 84-89, <https://doi.org/10.1109/CIMPS54606.2021.9652761>.
- Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps, *IEEE Software*, 33(3), 94-100.

- Forsgren, N., & Kersten, M. (2018). DevOps metrics. *Communications of the ACM*, 61(4), 44-48. <https://doi.org/10.1145/3159169>
- García-Mireles, G. A., Moraga, M. Á., García, F., & Piattini, M. (2015). Approaches to promote product quality within software process improvement initiatives: a mapping study. *Journal of Systems and Software*, 103, 150-166.
- Garousi, V., Tarhan, A., Pfahl, D., Coşkunçay, A., & Demirörs, O. (2019). Correlation of critical success factors with success of software projects: an empirical investigation. *Software Quality Journal*, 27, 429-493.
- Ghosheh, E., Black, S., & Qaddour, J. (2008). Design metrics for web application maintainability measurement. In *2008 IEEE/ACS International Conference on Computer Systems and Applications* (pp. 778-784). IEEE.
- Gunawan, F., & Budiardjo, E. (2021). A quest of software process improvements in DevOps and Kanban: A case study in small software company. In *Proceedings of the 2021 4th International Conference on Software Engineering and Information Management* (pp. 39-45).
- Humble, J. & Molesky, J. (2011). Why enterprises must adopt devops to enable continuous delivery. *Cutter IT J.*, 24(8), 6-12.
- ISO (2011) ISO/IEC 25010: Systems and software engineering – system and software product quality requirements and evaluation (SQuARE) – System and software quality models.
- Kato, D., Shimizu, A., & Ishikawa, H. (2022). Quality classification for testing work in DevOps. In *Proceedings of the 14th International Conference on Management of Digital EcoSystems* (pp. 156-162).
- Kitchenham, B. & Pfleeger, S. L. (1996). Software quality: the elusive target, *IEEE software*, 13(1), 12-21.
- Kitchenham, B. & Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering (EBSE 2007-001). Keele University and Durham University Joint Report.
- Lazuardi, M., Raharjo, T., Hardian, B. & Simanungkalit, T. (2021). Perceived Benefits of DevOps Implementation in Organization: A Systematic Literature Review. In *2021 10th International Conference on Software and Information Engineering, ICSIE 2021* (pp. 10-16). Association for Computing Machinery. <https://doi.org/10.1145/3512716.3512718>.
- Mallouli, W., Cavalli, A. R., Bagnato, A., & De Oca, E. M. (2020). Metrics-driven DevSecOps. In *ICSOFT* (pp. 228-233).
- Mishra, A. & Otaiwi, Z. (2020). DevOps and software quality: A systematic mapping, *Computer Science Review*, 38, 100308.

- Mohan, V. & Othmane, L. B. (2016). SecDevOps: Is It a Marketing Buzzword? - Mapping Research on Security in DevOps. *2016 11th International Conference on Availability, Reliability and Security (ARES)*, Salzburg, Austria, pp. 542-547, <https://doi.org/10.1109/ARES.2016.92>.
- Mohan, V., ben Othmane, L., & Kres, A. (2018). BP: Security concerns and best practices for automation of software deployment processes: An industrial case study. *In 2018 IEEE Cybersecurity Development (SecDev)* (pp. 21-28). IEEE.
- Moyón, F., Soares, R., Pinto-Albuquerque, M., Mendez, D., & Beckers, K. (2020). Integration of security standards in devops pipelines: An industry case study. *In Product-Focused Software Process Improvement: 21st International Conference, PROFES 2020*, (pp. 434-452). Springer International Publishing.
- Offerman, T., Blinde, R., Stettina, C. J., & Visser, J. (2022). A Study of Adoption and Effects of DevOps Practices. *In 2022 IEEE 28th International Conference on Engineering, Technology and Innovation (ICE/ITMC) & 31st International Association For Management of Technology (IAMOT) Joint Conference* (pp. 1-9). IEEE.
- Peña Olivero, N., Ávila George, H. & García-Mireles, G., (2023). Impacto de las prácticas de DevOps en la calidad de producto de software: hallazgos preliminares de un mapeo sistemático. *12 International Conference on Software Process Improvement (CIMPS 2023)*.
- Perera, P., Bandara, M., & Perera, I. (2016). Evaluating the impact of DevOps practice in Sri Lankan software development organizations. *In 2016 sixteenth international conference on advances in ict for emerging regions (icter)* (pp. 281-287). IEEE.
- Prates, L., Faustino, J., Silva, M., & Pereira, R. (2019). Devsecops metrics. *In Information Systems: Research, Development, Applications, Education: 12th SIGSAND/PLAIS EuroSymposium 2019*, (pp. 77-90). Springer International Publishing.
- Rahman, A. A. U., & Williams, L. (2016). Software security in DevOps: synthesizing practitioners' perceptions and practices. *In Proceedings of the international workshop on continuous software evolution and delivery* (pp. 70-76).
- Snyder, B., & Curtis, B. (2017). Using analytics to guide improvement during an Agile-DevOps transformation. *IEEE Software*, 35(1), 78-83.
- Tamburri, D. A., Bersani, M. M., Mirandola, R., & Pea, G. (2018). Devops service observability by-design: Experimenting with model-view-controller. *In Service-Oriented and Cloud Computing: 7th IFIP WG 2.14 European Conference, ESOC 2018*, (pp. 49-64). Springer International Publishing.
- Tomas, N., Li, J., & Huang, H. (2019). An empirical study on culture, automation, measurement, and sharing of devsecops. *In 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)* (pp. 1-8). IEEE.

- Visser, T., & Hulstijn, J. (2023). Measuring Agile/DevOps team performance. *In Agil-ISE@ CAiSE* (pp. 17-23).
- Wilkes, B., Milani, A. M. P., & Storey, M. A. (2023). A Framework for Automating the Measurement of DevOps Research and Assessment (DORA) Metrics. *In 2023 IEEE International Conference on Software Maintenance and Evolution (ICSME)* (pp. 62-72). IEEE.
- Zeller, M. (2021). Towards continuous safety assessment in context of devops. *In Computer Safety, Reliability, and Security. SAFECOMP 2021 Workshops: DECSoS, MAPSOD, DepDevOps, USDAI, and WAISE* (pp. 145-157). Springer International Publishing.

# Factores Sociales y Culturales que Influyen en las Técnicas de Priorización de Requisitos de Software: Un Estudio Secundario

Raúl Antonio Aguilar Vera<sup>1</sup>, Riad Aissrad Santos Charuf<sup>1</sup>,  
Julio C. Díaz Mendoza<sup>1</sup>, Juan Pablo Ucan Pech<sup>1</sup>

[avera@correo.uady.mx](mailto:avera@correo.uady.mx); [A16016280@alumnos.uady.mx](mailto:A16016280@alumnos.uady.mx); [julio.diaz@correo.uady.mx](mailto:julio.diaz@correo.uady.mx);  
[juan.ucan.uady.mx](mailto:juan.ucan.uady.mx)

<sup>1</sup> Facultad de Matemáticas, Universidad Autónoma de Yucatán, C.P. 97000, Mérida, Yucatán, México.

DOI: 10.17013/risti.53.53-68

**Resumen:** La tarea de priorización de requisitos resulta crítica para realizar un correcto proceso de requisitos software, y dada su naturaleza altamente social, es menester tener claridad de los factores sociales y culturales que inciden en dicho proceso. El presente estudio tiene como objetivo elaborar un estado del arte sobre los factores sociales y culturales, así como las técnicas vinculadas con la tarea de priorización de requisitos en el contexto del desarrollo de software. Para ello se especificaron 4 preguntas de investigación que sirvieron para guiar una revisión sistemática de literatura con la que se seleccionaron y analizaron 54 estudios primarios obtenidos de cuatro Bases de Datos y un repositorio. Entre los hallazgos se encontraron cinco categorías de factores sociales y culturales: comunicación, entendimiento del dominio, entendimiento de los stakeholders, influencias culturales y sociales, y manejo de múltiples stakeholders; así mismo, entre las técnicas que consideran dichos factores se pudieron identificar tres categorías: centradas en el usuario, centradas en patrones y basadas en la negociación, siendo la primera categoría la que concentra el mayor número de técnicas encontradas.

**Palabras-clave:** Factores sociales y culturales, Priorización de requisitos, Desarrollo de Requisitos de Software.

## *Social and Cultural Factors Influencing Software Requirements Prioritization Techniques: A Secondary Study*

**Abstract:** The task of prioritizing requirements is critical to carrying out a correct software requirements process, and given its highly social nature, it is necessary to be clear about the social and cultural factors that affect said process. The objective of this study is to develop a state of the art on social and cultural factors, as well as the techniques linked to the task of prioritizing requirements in the context of software development. To this end, 4 research questions were specified that served to guide a systematic literature review with which 54 primary studies obtained from four databases and a repository were selected and analyzed. Among the findings were five categories of social and cultural factors: communication, domain understanding, stakeholder understanding, cultural and social influences,

and management of multiple stakeholders; Likewise, among the techniques that consider these factors, three categories could be identified: user-centered, pattern-centered, and negotiation-based, with the first category concentrating the largest number of techniques found.

**Keywords:** Social and Cultural Factors, Software Requirements Development, Requirements Prioritization.

## 1. Introducción

La industria del Software ha crecido exponencialmente en el presente siglo, diversas tareas y actividades se han ido automatizando a un ritmo acelerado lo cual ha generado una gran demanda por disponer de software con prestaciones innovadoras; lo anterior, ha generado como resultado avances significativos en la investigación y estandarización de las diferentes tareas y actividades del proceso software. En el contexto de la primera fase del desarrollo software, la tarea de priorización de requisitos resulta crítica para realizar un correcto proceso de requisitos software, y dada su naturaleza altamente social, el Analista de Requisitos requiere interactuar con los interesados del sistema, por lo cual es deseable disponer del conocimiento vinculado con los factores sociales y culturales que impactan a dicha actividad. En la literatura, han sido identificado diversos factores sociales y culturales (Morales & Vega, 2018), para los cuales existen estudios que discuten en qué consisten, así como propuestas para gestionarlos, no obstante, dichos estudios al encontrarse aislados y por ende, tener un alcance limitado ofrecen conclusiones que no son generalizables.

En el contexto del proceso de requisitos, la construcción de un cuerpo de conocimientos ha avanzado a pasos agigantados con investigaciones que buscan contribuir a dicha fase del proceso de desarrollo software, en particular, sobre la tarea de priorización de requisitos (Alsanoosy, Spichkova & Harland, 2018; Hujainah et al, 2018; Riegel & Doerr, 2015; Sher et al, 2014; Thakurta, 2017). Entre los hallazgos reportados, los autores identificaron una carencia de estudios secundarios que ofrezcan un panorama general de los aspectos sociales y culturales involucrados en tarea de priorización de requisitos.

El presente artículo describe la Revisión Sistemática de Literatura (RSL) realizada para establecer un estado del arte respecto de factores sociales y culturales vinculados con la tarea de priorización de requisitos en el contexto de la fase de desarrollo de requisitos software. El artículo se organiza de la siguiente manera: la segunda sección presenta un breve marco teórico y un conjunto de estudios secundarios previos vinculados la tarea de priorización de requisitos; la sección 3 describe brevemente la metodología adoptada como guía para el desarrollo del estudio que se describe. Las secciones cuatro y cinco describen las fases de planificación y ejecución del estudio, de acuerdo con la metodología citada en la tercera sección. En la sexta se describen los resultados a las Preguntas de Investigación planteadas para el presente estudio. Finalmente, en la sección siete se presentan las conclusiones del estudio.

## 2. Antecedentes

La fase de desarrollo de requisitos, en el contexto del proceso software, integra un conjunto de tareas vinculadas con la obtención, análisis, especificación, y validación de

los requisitos software (Aguilar, Oktaba y Juárez, 2019); la priorización forma parte de las tareas de la fase de análisis de requisitos; ésta consiste en asignar prioridad a cada uno de los requisitos, pudiéndose clasificar en esenciales, útiles y deseables, siendo los primeros los de mayor prioridad (Hooks, 1993). El objetivo de la tarea de priorización, puede responder a diversos intereses, Sommerville y Sawyer (1997) señalan que dicha tarea tiene como objetivo el determinar la necesidad relativa de los requisitos, con la intención de identificar cuáles son los más críticos para el proyecto, lo anterior, permite refinar el documento de especificación de requisitos, de requisitos potencialmente innecesarios.

Con el propósito de analizar la pertinencia de un nuevo estudio secundario sobre la temática de priorización de requisitos, se realizó una revisión no sistemática identificando cinco estudios secundarios (Alsanoosy, Spichkova & Harland, 2018; Hujainah et al, 2018; Riegel & Doerr, 2015; Sher et al, 2014; Thakurta, 2017), los cuales sirvieron como antecedentes de nuestro estudio. Por cuestiones de espacio, describiremos solo dos de los estudios secundarios analizados.

Sher et al (2014) analizan diferentes aspectos a considerar durante la priorización de requisitos, sobre todo los aspectos del negocio y del cliente, y como las técnicas de priorización actuales los toman en consideración. Las preguntas de investigación establecidas para guiar el estudio son (1) ¿Qué aspectos de los requisitos han sido reportados y validados empíricamente para el proceso de priorización de requisitos de software? y (2) ¿Qué aspectos del negocio o cliente han sido reportados y empíricamente validados para la priorización de requisitos?”. Fueron consultadas siete bases de datos diferentes, estas fueron ACM, IEEE Xplore, Google search, CiteSeer, Wiley inter science, ISI web of knowledge y Metapress y, de estas bases de datos, se utilizaron 45 referencias en el estudio final. Como conclusión se obtuvo que la mayoría de las técnicas de priorización actuales no toman en consideración estos aspectos y que, por lo tanto, existe necesidad de investigar a profundidad las técnicas de priorización, así como la manera en que éstas toman en cuenta estos aspectos.

Por su parte, Alsanoosy, Spichkova & Harland (2018) exploran la influencia de la cultura de un lugar con respecto a las actividades de la ingeniería de requisitos, con base en lo reportado por estudios existentes. En este estudio se planteó como pregunta de investigación “¿Cómo influencia la cultura las actividades del proceso de requisitos?”, se consultaron las bases de datos IEEE Xplore, Springer, ACM Digital Library, Science Direct y Scopus, de las cuales se sustrajeron ochenta y un referencias. Los autores de dicho estudio identificaron dieciséis características culturales que influyen las actividades de ingeniería de requisitos.

### **3. Metodología de Investigación**

El presente es un estudio secundario en el que se utilizó como referencia la guía propuesta por Kitchenham & Charters (2007) para realizar Revisiones Sistemáticas de Literatura (RSL). Para incrementar el conjunto de estudios primarios relevantes, los autores complementaron el proceso de selección con la técnica bola de nieve hacia adelante (Wohlin, 2014).

## 4. Planificación del Estudio

### 4.1. Formulación de las Preguntas de Investigación

De acuerdo con el marco teórico del tema y con base en el análisis de los estudios secundarios previos encontrados, los autores especificaron un conjunto de Preguntas de Investigación (PI), las cuales sirvieron para guiar el presente estudio:

- *PI1. ¿Cuál ha sido la tendencia en investigación en el área de priorización de requisitos software durante el presente siglo?*
- *PI2. ¿Cuáles son los métodos de investigación utilizados por los estudios realizados durante el presente siglo en el contexto de la investigación en priorización de requisitos durante el presente siglo?*
- *PI3. ¿Cuáles son los factores sociales y culturales de los stakeholders considerados en la priorización de requisitos durante el presente siglo?*
- *PI4. ¿Cuáles son las técnicas de priorización de requisitos que consideran factores sociales o culturales en su implementación, en los estudios realizados durante el presente siglo?*

### 4.2. Definición de la Estrategia de Búsqueda

Con base en el marco teórico y el análisis de los estudios secundarios previos, se seleccionaron como fuentes de información: *IEEE Xplore*, *ACM Digital Library*, *ScienceDirect* y *SpringerLink*; adicionalmente, con el propósito de identificar estudios relevantes de otras fuentes no consideradas se decidió agregar el repositorio *Google Scholar*. De acuerdo con las Preguntas e Investigación (PI) acordadas, se seleccionó un conjunto limitado de palabras clave que pudieran servir para identificar estudios primarios pertinentes con la temática del estudio: factores (y conceptos asociados), requisitos, priorización, software. Con base en las palabras clave, se construyeron un conjunto de cadenas de búsqueda, las cuales serían utilizadas en la fase de ejecución. Las cadenas desarrolladas son:

- *CB1. Social AND (factors OR Issues OR Problems OR Aspects) AND Software AND Requirements AND prioritization*
- *CB2. Software AND Requirements AND Prioritization AND (“Social Issues” OR “Social aspects” OR “Cultural Considerations”)*
- *CB3. Software AND Requirements AND Prioritization AND (Culture OR Social OR Entomology OR Sociology)*

### 4.3. Criterios de Selección

Para seleccionar los estudios primarios vinculados con la temática y los que realmente proporcionan información relevante, en función de las PI, se establecieron un conjunto de criterios de selección, clasificados en Criterios de Inclusión (CI) y Criterios de Exclusión (CE). Los criterios acordados son los siguientes:

*Criterios de inclusión:*

- *CI1. Estudios publicados en revistas especializadas o en memorias de congresos.*

- CI2. Estudios publicados entre el año 2000 y el 2022.
- CI3. Estudios de investigación empírica que traten de forma específica los aspectos o factores sociales ligados a la priorización de requisitos.
- CI4. Estudios publicados en idioma inglés.

Criterios de exclusión:

- CE1. Estudios duplicados (se mantiene la fuente original).
- CE2. Estudios de los que no se tuvo acceso al contenido.

4.4.Técnica de Bola de Nieve hacia adelante

Para incrementar y mejorar el número de estudios primarios pertinentes, se decidió utilizar la técnica denominada bola de nieve hacia adelante (Wohlin, 2014), dicha técnica sería aplicada al el primer conjunto de estudios seleccionado; como herramienta para asistir dicha técnica se propone el uso de Google Scholar, el cual es un motor de búsqueda de contenido y bibliografía científico-académica permite identificar, con cierta vigencia, los artículos que han citado a una publicación.

4.5.Mecanismo de Evaluación de la Calidad de los EPS

El mecanismo de evaluación de la calidad en una RSL es imperativo para garantizar la validez y confiabilidad de los hallazgos descritos en los EPS, aparte de que permite ponderar la importancia de cada uno de los n estudios (Kitchenham, Budgen & Brereton, 2015). Ante la diversidad de metodologías de investigación utilizadas en los estudios empíricos en el contexto de la Ingeniería de Software, se decidió a adaptar el instrumento propuesto por Dyba & Dingsøyr (2008), obteniendo uno integrado por nueve ítems; dicho instrumento considera los criterios: (1) Calidad del informe, (1) Rigor, (3) Relevancia y (4) Credibilidad. La descripción de los criterios así como el instrumento resultante luego de su adaptación a nueve ítems, es descrito en mayor detalle en (Santos et al, 2024).

5. Ejecución del Estudio

5.1. Selección de los estudios primarios

Para la selección de los estudios primarios, se configuraron las cadenas de búsqueda elaboradas de acuerdo con los gestores de las bases de datos. Posteriormente se aplicaron los criterios de selección (CI/CE) obteniendo un total de treinta y nueve estudios primarios; la Tabla 1 ilustra el proceso por cada base de datos en las sus tres etapas.

| Base de Datos  | Cadena Configurada | CI | CE |
|----------------|--------------------|----|----|
| IEEE Xplore    | 20                 | 9  | 9  |
| ACM-DL         | 232                | 6  | 5  |
| ScienceDirect  | 133                | 13 | 10 |
| SpringerLink   | 357                | 16 | 13 |
| Google Scholar | 3,890              | 4  | 2  |

Tabla 1 – Número de EPS por fuente en cada etapa del proceso

Para completar el proceso de selección, se administró la técnica de bola de nieve hacia adelante al conjunto de treinta y nueve estudios primarios, identificando una grupo adicional de quince artículos pertinentes para el estudio, obteniendo de esta manera, un total de cincuenta y cuatro EPS.

5.2. Evaluación de la calidad de los EPS

La evaluación de la calidad para el conjunto de 54 EPS, se sintetiza en la información proporcionada en la Tabla 2; en términos generales, podemos concluir que la calidad de los EPS es pertinente para nuestro estudio.

| Criterio            | Valoración     |             |
|---------------------|----------------|-------------|
|                     | Cuantitativa   | Cualitativa |
| Calidad del Informe | 92/108 → 85%   | Perfecto    |
| Rigor               | 134/162 → 83%  | Perfecto    |
| Credibilidad        | 99.5/162 → 61% | Bueno       |
| Relevancia          | 51.5(54 → 93%  | Perfecto    |

Tabla 2 – Valoración de la Calidad del Conjunto de EPS

6. Resultados del Estudio

Con base en el análisis de los 54 EPS, las respuestas a las preguntas investigación son las siguientes:

PI1. ¿Cuál ha sido la tendencia en investigación en el área de priorización de requisitos software durante el presente siglo?

Desde principios de siglo y hasta el año 2015, se mantuvo poca producción sobre la temática de interés, en promedio un artículo por año; sin embargo, de 2015 a la fecha se observa un incremento importante en el interés en investigación sobre el tema de los factores sociales y culturales en la priorización de requisitos, en promedio cinco artículos por año. La Figura 1 ilustra la distribución de las publicaciones por año en la ventana de tiempo considerada.

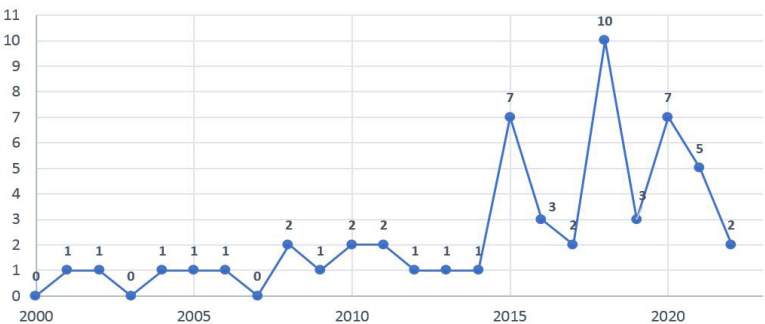


Figura 1 – Frecuencia de los EPS por año de publicación en la ventana de tiempo

*PI2. ¿Cuáles son los métodos de investigación utilizados por los estudios realizados durante el presente siglo en el contexto de la investigación en priorización de requisitos durante el presente siglo?*

De interés para los investigadores resultaba el conocer el tipo de investigación empírica realizada, a efecto de identificar el grado de profundización en el conocimiento de la temática. En la Figura 2 se puede observar que el Estudio de Caso es la técnica más recurrida con 21 de los 54 estudios, seguida por el Método Experimental con 20 estudios y en un tercer muy lejano, la Encuesta con 11 estudios. De acuerdo con Robson (2011) los niveles en los que se encuentra la investigación en tono a la priorización de requisitos, se presenta en los niveles: exploratoria, descriptiva y causa-efecto; con excepción de dos estudios que declaran el uso de Investigación-Acción con el propósito de mejora.

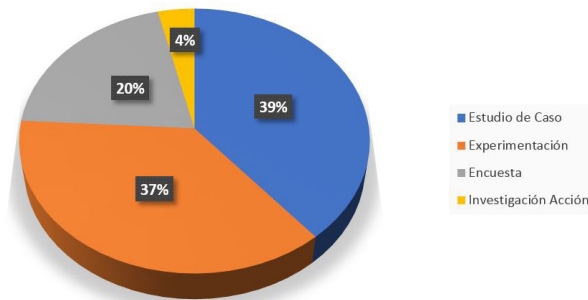


Figura 2 – % de EPS clasificados por método empírico utilizado

*PI3. ¿Cuáles son los factores sociales y culturales de los stakeholders considerados en la priorización de requisitos durante el presente siglo?*

Con base en el análisis de factores sociales y culturales citados de manera explícita o implícita con algún grado de influencia al proceso de priorización en cada uno de los 54 EPS, los autores proponen una tipología integrada por cinco categorías: (a) comunicación, (b) entendimiento del dominio (c) entendimiento de los stakeholders, (d) influencias sociales y culturales, (e) el manejo de múltiples stakeholders, así como (f) métodos mixtos. La Figura 3 ilustra el % de estudios que hacen referencia a cada uno de dichos factores.

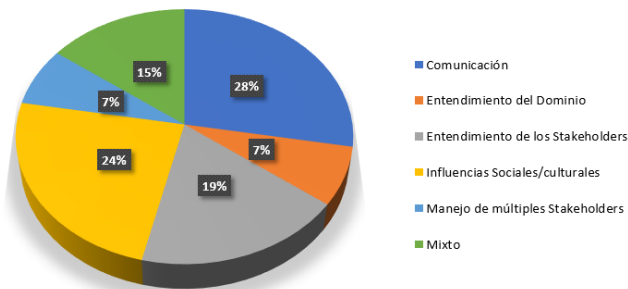


Figura 3 – % de EPS clasificados por factor social/cultural influyente

### A. Comunicación

Se refiere a la forma en la que debemos comunicarnos con los stakeholders, y se encuentra en función con las costumbres de la cultura dentro de la que se desempeña esta actividad, los medios por los cuales nos podemos comunicar con ellos para generar un sentimiento de inclusión durante el proceso de requisitos y las consideraciones necesarias respecto a su selección.

En el estudio de Zowghi (2018) se presentaron siete problemas vinculados con la comunicación, siendo estos:

1. La motivación de los usuarios para llevar a cabo el proceso de requisitos, ya que la falta de la misma afecta en gran medida las actividades de elicitación de requisitos, de negociación y de priorización debido a que estos pueden no tener una comunicación efectiva con los ingenieros de software.
2. La actitud, debido a que la falta de la misma, al igual que en el caso de la motivación, afecta en la manera en la que los stakeholders se comunican con los ingenieros y, por tanto, en los resultados de dicha comunicación.
3. Las expectativas, es imperativo comunicar y pedir que nos comuniquen las expectativas de los stakeholders respecto al sistema que se va a desarrollar, para poder definir con base en esto las prioridades de los requisitos y también para evitar desarrollar un sistema el cual no maneje estas de forma efectiva.
4. El expertise, ya que para llevar una comunicación efectiva con los stakeholders es importante saber con qué tantos tecnicismos o que tan específicamente se pueden tratar los temas referentes al dominio dentro del sistema.
5. La aceptación o resistencia al cambio, esto se debe considerar debido a que una mayor aceptación de los stakeholders al cambio llevará a que nos comuniquen o que podamos comunicar como prioridad los requisitos que brinden cambios más disruptivos dentro de la organización o sociedad dentro de la que se desempeñará el sistema, mientras que una mayor resistencia al cambio tendrá el efecto contrario.
6. La cooperación, esta problemática define la intención de los stakeholders de colaborar con los ingenieros durante el proceso de requisitos, comunicando de forma más o menos efectiva sus necesidades y prioridades con respecto al sistema.
7. Percepción del usuario del equipo de desarrollo, esto impacta grandemente la capacidad del equipo de desarrollo de llevar una comunicación efectiva con los stakeholders ya que una mala percepción del equipo por parte de estos llevará a una subestimación del mismo y, como resultado, a una falta de colaboración, de motivación y el tener una peor actitud durante la comunicación. Por lo tanto, es imperativo crear una buena percepción del equipo de desarrollo por parte de los stakeholders.

Por otro lado, estudios realizados por Bjarnason y Sharp (2015), Ehn et al (2021) y Derntl et. al (2015) se menciona la importancia de la distancia Geográfica como un problema para la comunicación con los stakeholders, el cual impacta tanto al agendar reuniones entre los ingenieros y los stakeholders, como en la elección del medio de comunicación con los mismos.

### B. *Entendimiento del Dominio*

Por entendimiento del dominio consideramos el conocimiento que se debe tener respecto al dominio de la problemática que el sistema abordará, por lo que por esta clasificación se considerarán los roles e interacciones de los diferentes stakeholders dentro del sistema y el entendimiento del ecosistema social en general del dominio del sistema. Dentro de los estudios realizados por Damian et al (2004); Lindgaard et al (2006) se menciona que no tener un correcto entendimiento del dominio, entendiendo por esto como el entendimiento de las necesidades objetivos y prioridades de la organización, suele ser causado por una mala comunicación durante las negociaciones con los stakeholders o un mal entendimiento de los diferentes roles de los stakeholders dentro de la organización. Por su parte Mussbacher et. al (2020) enfatizan la importancia de la consideración de los valores humanos dentro del modelado del dominio del sistema, entendiendo como valores humanos como el conjunto de virtudes que posee una persona u organización las cuales determinan el comportamiento e interacción con otros individuos y el espacio. En otro estudio, Duboc et al (2020b) argumentan que los sistemas de software, al ser impulsores importantes de cambios sociales, deben ser diseñados para mantener la sustentabilidad del amplio sistema socio-técnico al que va a ser integrado, por lo que se menciona la consideración de la sustentabilidad en el modelado del dominio; dichos autores identifican cinco dimensiones de la sustentabilidad que definen la capacidad del perdurar de un sistema socio-tecnológico: (1) Consideraciones ambientales, (2) Consideraciones económicas, refiriéndose al impacto económico que tendrá el sistema dentro de la sociedad en la que este se desempeñe, (3) Consideraciones sociales, (4) Consideraciones individuales, y (5) Consideraciones técnicas.

### C. *Entendimiento de los Stakeholders*

Se refiere al conocimiento que los ingenieros de software poseen respecto a los stakeholders, aspectos como: sus necesidades con respecto al sistema; su estilo de vida; su contexto socio-económico; su nivel de educación y conocimiento técnico del dominio; su edad, género y sus necesidades y expectativas emocionales con respecto al sistema. Rashid (2020) menciona la importancia de considerar la demografía de los stakeholders, argumenta que dependiendo de la demografía, el entendimiento y visualización del mundo y, por ende del problema, varía; comenta también que las prioridades y características preferentes de los stakeholders se encuentran correlacionados con su demografía, por lo que tener este entendimiento respecto a los stakeholders resulta imperativo. Otros estudios como el de Castro et al (2008), Richardson, Al-Ani y Ziv (2010), Miller et al (2015) y Mohabuth (2019) citan un conjunto de datos demográficos a considerar de los stakeholders como los siguientes: (1) Género, (2) Edad, (3) Educación, (4) Estado marital, (5) Localización geográfica, (6) Sueldo, (7) Rol dentro de la organización.

### D. *Influencias Culturales y Sociales*

Esta categoría agrupa las diferencias culturales y sociales entre los Ingenieros de Software y los stakeholders; la cultura del lugar donde se esté llevando a cabo el proceso de requisitos; la jerarquía dentro de la organización donde se esté llevando a cabo el proceso de requisitos; las influencias entre los stakeholders del sistema; y por último la dinámica social dentro de la que se ejecuta el proceso de requisitos.

Autores como (Alsanoosy et. al, 2018), (Brockmann & Thaumuller, 2009); (Alsanoosy, M. Spichkova & J. Harland, 2020a), (Marutschke, 2020), Thanasankit (2002) identificaron un conjunto de factores dentro de esta clasificación:

- El poder de la distancia: nos referimos a la jerarquía y su importancia dentro de la organización o cultura en la que se desempeña el proceso de requisitos, como puede ser el peso de la opinión de los gerentes dentro de la organización, la importancia de la opinión de las personas mayores dentro de las culturas conservadoras.
- Individualismo contra el colectivismo: Se refiere al peso que tienen las decisiones y prioridades del sistema con respecto a los beneficios personales frente a los beneficios colectivos que pueden brindar a la organización o sociedad dentro de la que se desempeñará el sistema.
- Resultados enfocados a largo o corto término: Este factor se refiere al enfoque al que se da importancia durante la priorización de requisitos en la cultura dentro de la que se esté llevando a cabo el proceso de requisitos, ya sea priorizando los resultados inmediatos o los resultados a largo plazo. Las culturas conservadoras suelen enfocarse en los resultados a largo plazo.
- Masculinidad contra la feminidad: Se refiere al poder de un género sobre el otro dentro de una cultura analizada, en particular, se refiere a la distribución de poder y de los roles entre ambos géneros.
- Evitación del conflicto y actitudes agresivas: Se refiere a que tan dispuestos están los stakeholders de entrar en conflictos o de tomar actitudes agresivas durante las negociaciones, toma de decisiones y en reuniones en general, lo anterior, con base en la cultura de la organización.

#### E. *Manejo de Múltiples Stakeholders*

Para la creación de sistemas a gran escala, o sistemas destinados a un público muy amplio, con stakeholders de diferentes contextos culturales y sociales, la actividad de priorización de requisitos se complica, ya que de manera inevitable los requisitos de varios stakeholders entrarán en conflicto o pueden resultar invisibilizados. Li et al (2018) y Santos et al (2016) analizan la gran variedad de stakeholders dentro del dominio del sistema, y mencionan la necesidad de agruparlos con base en características similares; estas agrupaciones se pueden realizar por medio de los datos demográficos obtenidos respecto a los stakeholders, la categorización de su jerarquía o importancia dentro del sistema, así como de su distribución geográfica, de esta forma se puede llevar a cabo un proceso de elicitación y de priorización de requisitos más ordenado y eficiente.

#### F. *Factores Mixtos*

Resulta importante comentar que los estudios cuyas problemáticas y factores sociales se clasifican como “mixtos” se refieren a los estudios que abordan dos o más de estas problemáticas o factores sociales dentro de sí mismos. Se identificaron en combinaron de dos factores: las influencias culturales y sociales con la comunicación, entendimiento del dominio con la comunicación, entendimiento del dominio con el entendimiento de los stakeholders; así mismo, en tres factores: entendimiento del dominio con Influencias culturales y sociales con comunicación, y por último, el entendimiento del dominio con el entendimiento de los stakeholders y con la comunicación. El entendimiento del

dominio, que de manera independiente solo fue identificado en el 7% de los estudios, como componente de factores mixtos, aparece en cuatro de las cinco combinaciones identificadas (6 de 8 estudios). Cabe mencionar que en un estudio previo (Santos et al, 2024) se analizan los EPS contrastando el método de investigación con el factor social y cultural involucrado.

*PI4. ¿Cuáles son las técnicas de priorización de requisitos que consideran factores sociales o culturales en su implementación, en los estudios realizados durante el presente siglo?*

En esta pregunta nos referimos a los métodos, frameworks o técnicas identificadas las cuales consideran dentro de su implementación los factores sociales previamente mencionados o que buscan solventar las diferentes problemáticas sociales identificadas previamente. Como resultado del análisis de los estudios encontrados se identificaron tres categorías diferentes: (1) Centradas en el usuario, (2) Centradas en patrones y (3) Basadas en la negociación. Las Tablas 3, 4 y 5 describen brevemente las técnicas identificadas.

| Técnica                              | Descripción                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Personas                             | Es una técnica que puede resultar muy útil ya que se crea una simulación de los stakeholders y, con base a esta simulación, se lleva a cabo el proceso de requisitos. Castro et al (2008) le encontraron debilidades, como que el 70% de los pasos de implementación se encuentran indefinidos o definidos de forma informal, por tanto los resultados son considerablemente informales, por que por sí sola no resulta una técnica muy viable para llevar a cabo las actividades de elicitación y priorización de requisitos.                                                                                                                                                                                                                                                                                                                                         |
| Sustainability awareness framework   | Duboc et al (2020b) proponen el framework Sustainability awareness framework (SuSAF), cuyo objetivo es el crear conciencia respecto a los efectos de la sustentabilidad que el sistema puede tener dentro del contexto en el que se desempeñará. El framework propone la creación de preguntas cuyo objetivo sea concientizar a los stakeholders y, con base en estas, realizar entrevistas con los mismos; dichas preguntas cubren las dimensiones: social, individual, ambiental, económica y técnica.                                                                                                                                                                                                                                                                                                                                                               |
| People oriented software engineering | Las técnicas o frameworks orientados a las emociones, abordar las emociones como uno de los impulsores principales de la definición de los requisitos, tomando en cuenta los objetivos y las necesidades emocionales de los stakeholders con respecto al sistema. En el estudio Curumsing et al (2019) se propone el modelo “People-oriented software engineering”; en dicho estudio se abordan las emociones personales, es decir, emociones que el stakeholder siente o que quiere sentir y no se encuentran ligadas al sistema; las emociones específicas del contexto, refiriéndose por esto a las emociones que el stakeholder siente o quiere sentir con respecto al sistema; y, por último, los objetivos emocionales, refiriéndose por esto a los objetivos no funcionales que describen una emoción a nivel reflectivo esperado de un rol dentro del sistema. |
| Affect grid                          | Colomo-Palacios et al (2011) propone la utilización del “affect grid” —una de las principales herramientas para medir las emociones humanas— para analizar y considerar las emociones de los stakeholders durante el proceso de requisitos. Esta herramienta es bastante simple en su implementación, ya que solamente busca saber cómo se sienten los diferentes stakeholders respecto a algún requisito funcional y, con base en ello, se calcula el sentimiento general respecto al requisito analizado.                                                                                                                                                                                                                                                                                                                                                            |
| GSD-RE                               | Ali & Lai (2016) proponen el método GSD-RE, el cual permite a los stakeholders obtener conciencia respecto a las diferencias geográficas, lingüísticas, temporales y sociales a considerar durante el proceso de requisitos y, como resultado de su utilización, se amplía el entendimiento del dominio del problema y obtienen requisitos de mayor calidad                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

| Técnica  | Descripción                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PHandler | <i>Priority Handler</i> (Phandler) es una técnica propuesta en Babar et al (2015), se basa la técnica de priorización de requisitos inteligente orientada al valor (VIRP), en una red neural y en el analytical hierarchical process (AHP) previamente mencionado en el estudio realizado por Saghir & Mustafa (2018).                                                                                                                                  |
| Macbeth  | Santos et al (2016) analizan la técnica <i>Measuring Attractiveness by the Categorical Based Evaluation Technique</i> (Macbeth). Consiste en un análisis basado en múltiples criterios, cada criterio es evaluado mediante la cuantificación de qué tan atractiva es una opción frente a las otras. En esta técnica recibe información del usuario y genera una jerarquía de que tan atractivos son los requisitos desde el punto de vista del usuario. |

Tabla 3 – Técnicas para priorización de requisitos “Centradas en el Usuario”

| Técnica                                 | Descripción                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Minado de Datos                         | Es una técnica con la que se trata de conseguir información relevante proveniente de una base de datos de gran tamaño. Debido a la gran cantidad de información que los stakeholders brindan durante el proceso de requisitos, la creación de nuevos medios de comunicación con los stakeholders y de obtención de requisitos como las redes sociales y las exigencias del mercado actuales este tipo de técnicas ha estado en auge en los últimos años. Nayeibi (2018) analiza la factibilidad de obtener nuevos requisitos y prioridades por medio del minado de datos en redes sociales y en tiendas virtuales de aplicaciones.                                                                  |
| Taxonomía GRADE                         | Papatheocharous et al (2018) proponen una técnica que ayuda a documentar la decisiones realizadas durante el proceso de requisitos para aprender de estas y comprender mejor las consecuencias de estas a futuro. La taxonomía GRADE analiza cinco ítems referentes a las decisiones, siendo estos los siguientes: (1) El objetivo que se busca alcanzar con la decisión, (2) Los roles involucrados en la toma de decisión, (3) Las opciones de decisión al momento de realizarla, (4) La manera en la que se decidió, refiriéndose por esto, de que manera se llegó al acuerdo de tomar esta decisión, y (5) El ambiente de la organización en la que esta decisión repercute.                    |
| Enfoque basado en términos lingüísticos | Gerogiannis & Tzikas (2017) crearon un enfoque, el cuál no nombraron, el cual consiste en que los stakeholders evalúan los requisitos de otros stakeholders utilizando términos lingüísticos para expresar esta evaluación y, con base en esto, obtener el nivel de importancia o relevancia de los requisitos. Este enfoque consiste en los siguientes pasos para su implementación: (1) Especificar el valor de importancia para los stakeholders y determinar el peso de los criterios de priorización, (2) Realizar la evaluación de los requisitos con respecto a los criterios de priorización y (3) Calcular las calificaciones brindadas de los stakeholders con respecto a los requisitos. |

Tabla 4 – Técnicas para priorización de requisitos “Centrados en Patrones”

## 7. Conclusiones y Trabajos Futuros

A partir del análisis de los 54 EPS se pudo identificar que de 2015 a la fecha se ha mantenido un interés significativo de investigación sobre la temática de los factores sociales y culturales en el contexto de la priorización de requisitos; así mismo, en cuanto al grado de profundización, con base en el tipo de estudio empírico realizado, es posible concluir que existe una variedad de estudios exploratorios, descriptivos y de análisis de relaciones causa-efecto, por lo que es posible afirmar que es necesario profundizar en la

investigación sobre la temática seleccionada mediante estudios de investigación-acción. En relación con los factores que inciden en el proceso de priorización de requisitos, se identificaron cinco categorías de factores culturales y sociales: (a) comunicación, (b) el entendimiento del dominio, (c) el entendimiento de los stakeholders, (d) las influencias culturales y sociales, y (e) el manejo de múltiples stakeholders. Siendo la (a), (c) y (d) las más utilizadas en los estudios analizados (71%). En cuanto a las técnicas, frameworks y métodos que consideran aspectos culturales y sociales para llevar a cabo las actividades de elicitación y priorización de requisitos, se encontró una tendencia actual por técnicas centradas en el usuario, donde hubieron más resultados y se encontró que existe cierta escasez por las basadas en la negociación con los stakeholders, por lo que realizar investigación respecto a este tema específico a futuro es de interés.

| Técnica               | Descripción                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Uso de redes sociales | Seyff et al (2015) y Mughal et al (2018) realizaron un análisis respecto a la creación de grupos de Facebook con los stakeholders, dentro de los cuales se lleva a cabo el proceso de elicitación y priorización de requisitos. Seyff et al (2015) realizaron un experimento donde se implementó este método de comunicación con los stakeholders con un bajo nivel de moderación y, como resultado, se encontró que la falta de moderación fue un factor decisivo en el fracaso de este método, debido a la falta de colaboración e interés de parte de los stakeholders con respecto al proceso de requisitos. En cuanto al nivel de moderación dentro de los grupos de Facebook, Mughal et al (2018) proponen un moderador como un requisito para la utilización de grupos de Facebook como medio de comunicación con los stakeholders y, apoyando este requisito de implementación |
| CaRE                  | En Elrakaiby et al (2021) se propone la métrica CaRE, que define los defectos existentes en los resultados de la comunicación con los stakeholders y propone operadores de refinamiento para disminuir el efecto de estos defectos. La utilización de CaRE propone una metodología de trabajo, donde los ingenieros consiguen un set inicial de requisitos durante el proceso y, con base en esto, se les pide a los stakeholders: (1) definir los tipos de defectos encontrados dentro de los requisitos y (2) asignarles operadores de refinamiento.                                                                                                                                                                                                                                                                                                                                 |

Tabla 5 – Técnicas para priorización de requisitos “Basadas en la Negociación”

Derivado de nuestro análisis, se encontró que de las técnicas actuales muy pocas logran solventar o abordar apropiadamente las problemáticas encontradas dentro del proceso de requisitos en desarrollos globales, donde los factores sociales específicos de este tipo de desarrollos no son considerados de forma apropiada, por lo que se concluye como tema de interés para investigaciones futuras.

Referencias

Aguilar, R., Oktaba, H. & Juárez R. (2019). Introducción a la Ingeniería de Software. En Aguilar, R. (ed.) Ingeniería de Software en México: Educación, Industria e Investigación. Academia Mexicana de Computación, (pp. 3-5).

Alsanoosy, T., Spichkova, M. & Harland, J. (2018). Cultural Influences on the Requirements Engineering Process: Lessons Learned from Practice. 2018 23rd International Conference on Engineering of Complex Computer Systems (ICECCS). <https://doi.org/10.1109/iceccs2018.2018.00015>.

- Alsanoosy, T., Spichkova, M., & Harland, J. (2020). Cultural influence on requirements engineering activities: a systematic literature review and analysis. *Requirements Engineering*, 25(3), 339-362.
- Bjarnason, E. & Sharp, H. (2015). The role of distances in requirements communication: a case study. *Requirements Engineering*, 22(1), 1-26.
- Brockmann, P. S. & Thaumuller, T. (2009). Cultural Aspects of Global Requirements Engineering: An Empirical Chinese-German Case Study. 2009 Fourth IEEE International Conference on Global Software Engineering.
- Castro, J., Acuña, S. & Juristo, N. (2008). Enriching requirements analysis with the Personas technique. Interplay between Usability Evaluation and Software Development.
- Colomo-Palacios, R. Casado-Lumbreras, C., Soto-Acosta, P. & García-Crespo, A. (2011). Using the Affect Grid to Measure Emotions in Software Requirements Engineering. CERN European Organization for Nuclear Research - Zenodo.
- Curumsing, M. K., Fernando, N., Abdelrazek, M., Vasa, R., Mouzakis, K. & Grundy, J. (2019). Emotion-oriented requirements engineering: A case study in developing a smart home system for the elderly. *Journal of Systems and Software*, 147, 215-229. <https://doi.org/10.1016/j.jss.2018.06.077>
- Damian, D., Zowghi, D., Vaidyanathasamy, L. & Pal, Y. (2004). An Industrial Case Study of Immediate Benefits of Requirements Engineering Process Improvement at the Australian Center for Unisys Software. *Empirical Software Engineering*, 9(1/2), 45-75.
- Derntl, M., Renzel, D., Nicolaescu, P., Koren, I. & Klamma, R. (2015). Distributed Software Engineering in Collaborative Research Projects. 2015 IEEE 10th International Conference on Global Software Engineering.
- Duboc, L., Penzenstadler, B., Porras, J., Akinli Kocak, S., Betz, S., Chitchyan, R., Leifler, O., Seyff, N. & Venters, C. C. (2020b). Requirements engineering for sustainability: an awareness framework for designing software systems for a better tomorrow. *Requirements Engineering*, 25(4), 469-492.
- Dyba, T., & Dingsøyr, T. (2008). Strength of evidence in systematic reviews in software engineering. *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*.
- Ehn, M., Derneborg, M., Revenäs, S. & Cicchetti, A. (2021). User-centered requirements engineering to manage the fuzzy front-end of open innovation in e-health: A study on support systems for seniors' physical activity. *International Journal of Medical Informatics*, 154, 104547.
- Elrakaiby, Y., Borgida, A., Ferrari, A. & Mylopoulos, J. (2021). CaRE: a refinement calculus for requirements engineering based on argumentation theory. *Software and Systems Modeling*, 21(6), 2113-2132.

- Gerogiannis, V. C., & Tzikas, G. (2017). Using Fuzzy Linguistic 2-Tuples to Collectively Prioritize Software Requirements based on Stakeholders' Evaluations. In Proceedings of the 21st Pan-Hellenic Conference on Informatics, pp. 1-6.
- Hooks, I. (1993). Writing Good Requirements, A requirements Working Group Information Report. Proceedings of the Third International Symposium of the INCOSE.
- Hujainah, F., Abu Bakar, R. B., Al-Haimi, B., & Abdulgaber, M. A. (2018). Stakeholder quantification and prioritisation research: A systematic literature review. Information and Software Technology, 102, 85–99.
- Kitchenham, B. A., Budgen, D., & Brereton, P. (2015). Assessing Study Quality. In Evidence-Based Software Engineering and Systematic Reviews, (pp. 79-88). CRC Press.
- Li, C., Huang, L., Ge, J., Luo, B., & Ng, V. (2018). Automatically classifying user requests in crowdsourcing requirements engineering. Journal of Systems and Software, 138, 108-123.
- Marutschke, D. M., Kryssanov, V. V. & Brockmann, P. (2020). Teaching Distributed Requirements Engineering: Simulation of an Offshoring Project with Geographically Separated Teams. 2020 IEEE 32nd Conference on Software
- Miller, T., Pedell, S., Lopez-Lorca, A. A., Mendoza, A., Sterling, L. & Keirnan, A. (2015). Emotion-led modelling for people-oriented requirements engineering: The case study of emergency systems. Journal of Systems and Software, 105, 54-71. <https://doi.org/10.1016/j.jss.2015.03.044>
- Mohabuth, A. (2019). Software Practitioners Challenges in the Requirement Engineering Phase of Software Development. PSENR-19, LCABES-19, L2HSS-19 & ABLE-19 June 19-21, 2019 Lisbon (Portugal).
- Morales, N., & Vega, V. (2018). Factores Humanos y la Mejora de Procesos de Software: Propuesta inicial de un catálogo que guíe su gestión. RISTI - Revista Ibérica de Sistemas y Tecnologías de Información, (29), 30-42.
- Mughal, S., Abbas, A., Ahmad, N., & Khan, S. U. (2018). A social network based process to minimize in-group biasedness during requirement engineering. IEEE Access, 6, 66870-66885.
- Mussbacher, G., Hussain, W. & Whittle, J. (2020). Is There a Need to Address Human Values in Domain Modelling? 2020 IEEE Tenth International Model-Driven Requirements Engineering (MoDRE).
- Nayebi, M., Cho, H., & Ruhe, G. (2018). App store mining is not enough for app improvement. Empirical Software Engineering, 23, 2764-2794.
- Papatheocharous, E., Wnuk, K., Petersen, K., Sentilles, S., Cicchetti, A., Gorschek, T., & Shah, S. M. A. (2018). The GRADE taxonomy for supporting decision-making of asset selection in software-intensive system development. Information and Software Technology, 100, 1-17.

- Rashid, A. (2020). Conceptualization of smartphone usage and feature preferences among various demographics. SpringerLink.
- Richardson, D., Al-Ani, B., & Ziv, H. (2010). Requirements engineering at the margins: avoiding technological hubris through alternative approaches. In *Proceedings of the FSE/SDP workshop on Future of software engineering research*, (pp. 303-308).
- Riegel, N., & Doerr, J. (2015). A systematic literature review of requirements prioritization criteria. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, (pp. 300-317). Springer, Cham.
- Robson, C. (2011). *Real World Research: A Resource for Social Scientists and Practitioner-Researchers*. 3a Ed, Wiley.
- Saghir, S. & Mustafa, T. (2018). Requirements Prioritization Techniques for Global Software Engineering. *Pakistan Journal of Engineering, Technology & Science*, 6(1). <https://doi.org/10.22555/pjets.v6i1.1143>
- Santos, R., Albuquerque, A., & Pinheiro, P. R. (2016). Towards the applied hybrid model in requirements prioritization. *Procedia Computer Science*, 91, 909-918.
- Santos, R., Aguilar, R., Díaz, J. & Uacán, J. (2024) Estudio Secundario sobre Factores Sociales que Inciden en la Priorización de Requisitos Software. *Memorias del Congreso Internacional de Mejora de Procesos 2023*. IEEE.
- Seyff, N., Todoran, I., Caluser, K., Singer, L. & Glinz, M. (2015). Using popular social network sites to support requirements elicitation, prioritization and negotiation. *Journal of Internet Services and Applications*, 6(1).
- Sher, F., Jawawi, D., Mohamad, R. & Babar, M. (2014). Requirements prioritization techniques and different aspects for prioritization a systematic literature review protocol. 8th. *Malaysian Software Engineering Conference*, 31-36.
- Sommerville, I. & Sawyer, P. (1997) *Requirements Engineering: A good practice guide*. WILEY.
- Thakurta, R. (2017). Understanding requirement prioritization artifacts: a systematic mapping. *Requirements Eng* 22, 491–526.
- Thanasankit, T. (2002). Requirements engineering—exploring the influence of power and Thai values. *European Journal of Information Systems*. <https://doi.org/10.1057/palgrave.ejis.3000423>
- Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th ICE & A in SE..*
- Zowghi, D. (2018). «Affects» of User Involvement in Software Development. 2018 1st *International Workshop on Affective Computing for Requirements Engineering (AffectRE)*. <https://doi.org/10.1109/affectre.2018.00008>

# Auditoría de riesgos de ciberseguridad: Revisión de Literatura, propuesta y aplicación

I.D. Sanchez-Garcia<sup>1</sup>, A.M. Rea-Guaman<sup>2</sup>, T. San Feliu<sup>1</sup> and J.A<sup>1</sup>. Calvo-Manzano<sup>1</sup>.

isaacdaniel.sanchez@alumnos.upm.es; amrea1@espe.edu.ec; tomas.sanfeliu@upm.es;  
joseantonio.calvomanzano@upm.es.

<sup>1</sup> Universidad Politécnica de Madrid, ETS Ingenieros Informáticos Madrid, España.

<sup>2</sup> Universidad de las Fuerzas Armadas – ESPE, Departamento de Ciencias de la Computación Sangolquí, Ecuador.

DOI: 10.17013/risti.53.69–87

**Resumen:** Una de las etapas de la gestión de riesgos de ciberseguridad es el monitoreo y la revisión. Esta etapa forma parte del proceso de mejora continua de un sistema de gestión de riesgos de ciberseguridad. Este artículo tiene como objetivo llevar a cabo una exploración de una guía de auditoría de riesgos de ciberseguridad, tomando como referencia objetivos comunes y guías de la auditoría de riesgos de ciberseguridad. Para ello se tomó como partida una Revisión Sistemática de Literatura (SLR) considerando los estudios de los últimos diez años (2012-2022), a partir de los cuales se identificaron 23 estudios que mencionaban objetivos y guías de auditoría de riesgos de ciberseguridad. Además, se propusieron atributos que deben ser considerados para la creación de un guía de riesgos de ciberseguridad. Posteriormente, en el presente trabajo se define una guía de auditoría de riesgos de ciberseguridad (CRAG). Finalmente, se expone la aplicación de CRAG por medio de un caso de estudio, considerando los parámetros identificados en los estudios previamente mencionados.

**Palabras-clave:** Auditoría de ciberseguridad, Guía de auditoría, Aseguramiento, Cumplimiento, Mejora de la ciberseguridad, Monitoreo, Revisión sistemática de literatura, Aplicación.

## *Cybersecurity Risk Audit: Literature Review, Proposal, and Application*

**Abstract:** One of the stages of cybersecurity risk management is monitoring and review. This stage is part of the continuous improvement process of a cybersecurity risk management system. This article aims to conduct an exploration of a cybersecurity risk audit guide by referencing common objectives and guidelines of cybersecurity risk auditing. To do so, a Systematic Literature Review (SLR) was conducted considering studies from the last ten years (2012-2022), from which 23 studies mentioning cybersecurity risk audit objectives and guidelines were identified. Additionally, attributes to be considered for the creation of a cybersecurity risk guide were proposed. Finally, an application and validation of the identified parameters in the previously mentioned studies are presented.

**Keywords:** Cybersecurity Audit, Audit Guide, Assurance, Compliance, Cybersecurity Enhancement, Monitoring, Systematic Literature Review, Implementation.

## 1. Introducción

La norma ISO 27005:2018 establece como marco de referencia para la gestión de riesgos de ciberseguridad cuatro etapas: 1) establecimiento del contexto, 2) evaluación de riesgos, 3) tratamiento de riesgos, 4) monitoreo y revisión. La nominación de estas etapas puede variar según el marco de referencia utilizado. Comúnmente, la etapa de monitoreo y revisión suele cerrar el ciclo de gestión de riesgos de ciberseguridad.

La etapa de monitoreo y revisión de riesgos de ciberseguridad incluye el monitoreo regular de los riesgos y contramedidas definidas en las etapas de evaluación y tratamiento de riesgos. Además, revisiones periódicas (auditorías) son necesarias para un monitoreo adecuado para validar la efectividad de las contramedidas en abordar los riesgos (Slapničar et al., 2022). Este estudio se centrará en los problemas relacionados con las auditorías de riesgos de ciberseguridad, que son parte de la etapa de monitoreo y revisión de riesgos de ciberseguridad.

La evaluación de riesgos de ciberseguridad implica examinar regularmente los mecanismos de seguridad para detectar deficiencias donde no hayan sido diseñados de manera efectiva o no estén operando como se esperaba. Estas evaluaciones pueden abarcar áreas como las aplicaciones de tecnologías de la información, las operaciones de TI, la gobernanza y el personal, todas las cuales son componentes específicos en la implementación de sistemas de información.

El estudio publicado por la Confederación Europea de Institutos de Auditores Internos (ECIIA) (European Confederation of Institutes of Internal Auditors, 2020) resalta la importancia de las auditorías de riesgos de ciberseguridad, colocándolas como la principal preocupación de los Chief Audit Executives (CAE, por sus siglas en inglés). Diversos autores como Slapničar et al., (2022) y Duncan & Whittington, (2014) coinciden en que la auditoría de riesgos de ciberseguridad es un campo relativamente nuevo.

A pesar de su importancia, la auditoría de riesgos de ciberseguridad aún no ha alcanzado un nivel de madurez que pueda ser comparable respecto a otras áreas de auditoría, como lo es la auditoría contable, que se ha venido desarrollando durante décadas. Esta falta de madurez es justificable, pues la ciberseguridad es un campo que está en constante evolución, con nuevos riesgos y amenazas surgiendo continuamente. Autores como Duncan y Whittington (2014) enfatizan en la necesidad de avanzar en este sentido.

Para que la auditoría de riesgos de ciberseguridad alcance un nivel de madurez considerado óptimo, se requiere de una mayor claridad en tres aspectos fundamentales, que son los conceptos, las pruebas y los procedimientos. Estos tres elementos son requeridos para que el correcto funcionamiento de las medidas de seguridad pueda ser validado.

Los desafíos principales, a los cuales se enfrentan las organizaciones al momento de realizar las auditorías de riesgo de ciberseguridad son las siguientes:

- a. La constante aparición de nuevas amenazas y vulnerabilidades representan un obstáculo para llevar a cabo auditorías de ciberseguridad.
- b. La escasez de profesionales que posean las habilidades y conocimientos para realizar auditorías de riesgo de ciberseguridad es otro problema común. Muchas organizaciones recurren a la ayuda de consultores externos cualificados para tomar decisiones estratégicas en este ámbito (Galligan & Rau, 2015).

Por otro lado, las organizaciones también enfrentan un reto al intentar cumplir con un estándar o guía de auditoría de riesgos de ciberseguridad. De acuerdo con Sabillón et al. (2018), es crucial contar con una orientación clara o un consenso sobre qué áreas, subáreas, dominios o subdominios deben ser considerados en una auditoría de riesgos de ciberseguridad. Un ejemplo que ilustra las dificultades asociadas con los estándares actuales de auditoría de riesgos de ciberseguridad es el estudio realizado por Gauthier & Brender, (2021).

Un estudio realizado por Megan et al. (2022) recopila datos de diversos estudios relacionados con las auditorías de riesgos de ciberseguridad. El estudio revela que las organizaciones, especialmente en países en vías de desarrollo, se encuentran realizando esfuerzos significativos para mejorar la ejecución de estas auditorías.

Aquellas organizaciones que han experimentado problemas de ciberseguridad por la falta de auditorías de riesgos de ciberseguridad han sido las promotoras impulsoras de este cambio, pues han reconocido la importancia de estas auditorías para de esta manera, ser capaces de proteger sus activos de información (Gale, Bongiovanni, & Slapnicar, 2022).

Para ser prácticos, los objetivos de la auditoría de riesgos de ciberseguridad deben estar alineados desde un punto de vista de control interno (Sabillon et al., 2018). Diversos estudios de investigación, como los realizados por Steinbart et al. (2016) y Stafford et al. (2018) han analizado el papel fundamental que juega la auditoría interna en la gestión y gobernanza de riesgos de ciberseguridad. Estos estudios resaltan la importancia de que la auditoría interna se enfoque en evaluar la eficacia del control interno en materia de ciberseguridad.

Gale et al., (2022) recopilan datos de estudios relacionados con auditorías de riesgos de ciberseguridad. Como resultado, Gale et al., (2022) mencionan que las organizaciones están realizando esfuerzos significativos, especialmente en países en desarrollo, para mejorar la ejecución de las auditorías de riesgos de ciberseguridad. Las organizaciones que ya han experimentado problemas de ciberseguridad debido a la falta de auditorías de riesgos de ciberseguridad han trabajado en mejorar la competencia de los comités de auditoría interna e implementar guías prácticas dentro de la organización, definiendo objetivos claros (Gale et al., 2022).

Para ser prácticos, los objetivos relacionados con la auditoría de riesgos de ciberseguridad deben ser vistos desde un punto de vista de control interno (Sabillon et al., 2018). Algunos estudios de investigación (por ejemplo, (Steinbart et al., 2016); (Stafford et al., 2018)) han analizado el papel de la auditoría interna en la gestión y gobernanza de riesgos de ciberseguridad.

Los esfuerzos actuales para abordar las auditorías de riesgos de ciberseguridad se basan en el uso de guías y marcos de referencias conocidos. Entre las guías más utilizadas se

encuentran la familia ISO 27000, específicamente la ISO/IEC 27005 (2018), el NIST Cybersecurity Framework (CSF) (National Institute of Standards and Technology, 2018) y Control Objectives for Information and Related Technologies (COBIT) (ISACA, 2018). Para implementar efectivamente estas guías, es necesario integrar estos conceptos al campo del control interno para abordar la ciberseguridad de manera integral, no solo considerando los aspectos técnicos, sino también los procesos y las personas involucradas (Islam, Farah, & Stafford, 2022).

Según Ezzamouri & Hulstijn (2018), una auditoría de riesgos de ciberseguridad realizada con la guía adecuada permite a los auditores ofrecer una garantía razonable sobre la criticidad e impacto potencial de un riesgo. Esta garantía se basa en la evaluación de la efectividad de las contramedidas implementadas para mitigar dicho riesgo. Los resultados obtenidos de esta auditoría de riesgos de ciberseguridad se plasman en un informe de auditoría. El monitoreo continuo suele ser responsabilidad de la dirección, ya que afecta a los controles internos integrados en los procesos tecnológicos de la empresa.

Derivado de esta problemática relacionada con la auditoría de riesgos de ciberseguridad se decidió realizar una Revisión Sistemática de la Literatura (SLR, por sus siglas en inglés) utilizando como guía a Kitchenham et al., (2009) presentada en la sección 2. En la sección 3 se presenta la Guía de Auditoría de Riesgos de Ciberseguridad (CRAG, por sus siglas en inglés)

Se seleccionó la Técnica Estructurada de Análisis y Diseño (SADT) de Congram & Epelman, (1995) para su desarrollo. En la sección 4 se presenta validación de la guía CRAG, esta validación se realizó por medio de un caso de estudio. Finalmente, en la sección 5 se presentan las conclusiones y líneas de investigación futuras.

Las contribuciones de este trabajo de investigación serán:

1. Establecer objetivos comunes aplicables a las auditorías de riesgos de ciberseguridad.
2. Discutir las contribuciones y beneficios de las guías de auditoría de riesgos de ciberseguridad.
3. Realizar una propuesta de guía de auditoría de riesgos de ciberseguridad.
4. Realizar la validación de la guía previamente propuesta por medio de la aplicación de un caso de estudio.

## **2. Revisión sistemática de literatura**

La metodología propuesta por B. Kitchenham fue seleccionada para llevar a cabo esta Revisión Sistemática de la Literatura (SLR, por sus siglas en inglés) (Kitchenham et al., 2009). La SLR es un proceso formal y verificable que el investigador realiza para documentar el estado del conocimiento sobre un tema en particular. Según B. Kitchenham, una revisión sistemática evalúa e interpreta toda la investigación disponible relevante para una pregunta de investigación dada en el área temática o fenómeno de interés (Fernandez et al., 2015). Esta SLR incluye los siguientes pasos: (1) Proponer un protocolo de revisión y definir las preguntas de investigación, (2) Realizar la revisión (identificar estudios primarios y evaluar estos estudios), (3) Extraer y discutir los resultados de la revisión sistemática de la literatura.

## 2.1. Proponer un protocolo de revisión y definir las preguntas de investigación

**Pregunta de Investigación 1 (RQ1):** ¿Qué objetivos se utilizan en las auditorías de riesgos de ciberseguridad según la literatura?

Con el propósito de responder la pregunta de investigación RQ1, se identificarán los objetivos primordiales de las auditorías de riesgos de ciberseguridad que se encuentran plasmados en la literatura existente. Asimismo, estos objetivos serán clasificados según su enfoque específico (auditoría interna, seguridad de la información o ciberseguridad) con la finalidad de delimitar el conjunto de aquellos aplicables a la ciberseguridad. La formulación de la RQ1 se fundamenta en el objetivo a) identificar los principales objetivos de las auditorías de riesgos de ciberseguridad.

**Pregunta de Investigación 2 (RQ2):** ¿Cuáles son las pautas más utilizadas en la literatura sobre auditorías de riesgos de ciberseguridad y sus características?

El propósito fundamental de la RQ2 consiste en identificar las directrices, estándares, marcos o modelos que guarden relación con las auditorías de riesgos de ciberseguridad. Adicionalmente, persigue el objetivo de detectar tendencias o brechas en las mencionadas pautas y estándares. La formulación de la RQ2 se basa en el objetivo b) identificar y analizar las pautas de auditoría de riesgos de ciberseguridad.

### 2.1.1. Protocolo PICO

El protocolo PICO (población, intervención, comparación y resultado), establecido en el modelo de Kitchenham et al. (2009), el cual ha sido ampliamente utilizado y extensamente probado en la ejecución de revisiones sistemáticas de la literatura, fue seleccionado para su implementación en nuestro estudio.

- Población: Se consideran las publicaciones relacionadas con los objetivos de las auditorías de riesgos de ciberseguridad. Además, también se toman en cuenta las pautas vinculadas con las auditorías de riesgos de ciberseguridad.
- Intervención: Los estudios de investigación mencionan diferentes objetivos y pautas relacionados con las auditorías de riesgos de ciberseguridad.
- Comparación: Identificar y comparar objetivos y pautas relacionadas con las auditorías de riesgos de ciberseguridad mencionadas por las publicaciones.
- Resultado: Objetivos de las auditorías de riesgos de ciberseguridad, frecuencia y enfoque de las pautas de auditoría de riesgos de ciberseguridad.

### 2.1.2. Generación de cadena de búsqueda de investigación

Se ha llevado a cabo una cuidadosa selección de un conjunto de palabras clave con el objetivo de obtener los resultados más relevantes posibles. Con la finalidad de evitar la exclusión de estudios pertinentes y construir la cadena de búsqueda de manera eficiente, se han incluido sinónimos para las palabras clave. Dichos sinónimos, así como las palabras clave, se encuentran detallados en la Tabla 1.

Las cadenas de búsqueda han sido construidas utilizando las palabras clave y sus respectivos sinónimos presentes en la Tabla 1. Se han añadido los conectores lógicos

“AND” y “OR” para unir las palabras clave y sinónimos, dando lugar a la siguiente cadena de búsqueda genérica: (“Auditoría” OR “Aseguramiento” OR “Guía” OR “Ley” OR “Estándar”) AND (“Ciberseguridad” OR “Seguridad Cibernética” OR “Seguridad de la Información”).

| Palabra Clave        | Sinónimos                            |
|----------------------|--------------------------------------|
| <i>Cybersecurity</i> | Cyber security, Information security |
| <i>Audit</i>         | Assurance, Guide, Law, Standard      |

Tabla 1 – Palabras clave y sinónimos para construir las cadenas de investigación

**2.2. Realizar la revisión**

Los criterios establecidos para la selección de fuentes (bases de datos) se fundamentaron en: a) la experiencia reportada por Dyba et al. (2007), y Petersen et al. (2008); b) la disponibilidad de estudios de texto completo encontrados a través de las cadenas de búsqueda; c) estudios disponibles en la web de forma gratuita; y d) revistas especializadas en ciberseguridad accesibles en la biblioteca de la Universidad Politécnica de Madrid. Las fuentes seleccionadas para esta Revisión Sistemática de Literatura son 1) ACM Digital Library, 2) IEEE Digital Library, 3) ISI Web of Science, 4) ScienceDirect, 5) Scopus y 6) Springer Link. Google Scholar fue seleccionado únicamente para aplicar la técnica de bola de nieve.

La cadena de búsqueda definida en la sección anterior fue introducida en los motores de búsqueda de las fuentes mencionadas.

**2.2.1. Criterios de inclusión y exclusión**

Es necesario definir criterios de inclusión y exclusión (I&E) con el propósito de seleccionar los estudios relacionados con la Revisión Sistemática de Literatura. Los criterios de I&E fueron definidos en base a la experiencia de Kitchenham et al. (2009), y Petersen et al. (2008). La selección de estudios se ha llevado a cabo utilizando los siguientes criterios de I&E.

**Criterios de inclusión:**

- Bibliotecas digitales precisas: Se evaluó la calidad, cantidad y confiabilidad de los estudios publicados en siete bibliotecas digitales (seleccionadas según los criterios establecidos para la selección de fuentes).
- Contenido de los estudios: Se constató que el tema principal de los estudios identificados estaba relacionado con las preguntas de investigación PI1 y PI2. Para este análisis, se validaron el título, el resumen y las palabras clave de los estudios.
- Coherencia del estudio: Se confirmó que los estudios estuvieran relacionados con el campo de la ciberseguridad. Para ello, se identificaron el área y el título de los estudios, asegurando su coherencia con los objetivos de la Revisión Sistemática de Literatura.

- Estudios de texto completo: Se revisaron todos los estudios identificados en las bibliotecas digitales, validando la integridad de la información. De esta manera, únicamente se incluyeron estudios completos.

**Criterios de exclusión:**

- Estudios duplicados: Se eliminaron los estudios duplicados presentes en las diferentes bibliotecas digitales. Este criterio de exclusión tiene como finalidad reducir el volumen de información innecesaria.
- Estudios basados únicamente en una opinión particular: Se excluyeron los estudios que solo mencionaban una opinión particular. Este criterio de exclusión busca contar con estudios basados en hipótesis científicas validadas.
- Estudios anteriores a 2012: Debido a la continua actualización del campo de la ciberseguridad, se excluyeron los estudios previos a 2012. Por lo tanto, la información de un estudio anterior a 2012 se consideraría irrelevante para esta Revisión Sistemática de Literatura.
- Estudios irrelevantes para las preguntas de investigación o no relacionados con el tema: También se excluyeron los estudios no relacionados con ninguna de las preguntas de investigación PI1 y PI2.
- Estudios poco claros o ambiguos: Se descartaron los estudios que no aclaraban sus contribuciones o no especificaban su relación con la ciberseguridad.
- Estudios que mencionan auditoría de riesgos, pero no en el campo de la ciberseguridad: Se descartaron los estudios que mencionaban pautas de auditoría relacionadas con áreas distintas de la ciberseguridad.
- Literatura gris, o literatura publicada por editores no tradicionales: Este tipo de literatura fue excluido.

**2.3. Selección de estudios primarios**

La selección de los estudios primarios se ha dividido en un procedimiento compuesto por tres actividades. Se creó un conjunto de datos para realizar estas actividades (ver <https://short.upm.es/xljfp>). Este conjunto de datos contiene los resultados de las tres actividades.

En la primera actividad, se insertó la cadena de búsqueda creada en el protocolo de revisión en los motores de búsqueda de la base de datos. Después de la primera actividad, se encontraron 2,032 estudios. Los resultados se presentan en la Tabla 2.

| Fuente                         | Estudios<br>iniciales<br>(Primera<br>actividad) | Estudios<br>relevantes<br>(Segunda<br>actividad) | Estudios<br>primarios Tarea<br>1 (Tercera<br>actividad) | Estudios<br>primarios-Tarea<br>2 (Tercera<br>actividad) |
|--------------------------------|-------------------------------------------------|--------------------------------------------------|---------------------------------------------------------|---------------------------------------------------------|
| ACM Digital Library<br>(ACM)   | 464                                             | 12                                               | 4                                                       | 4                                                       |
| IEEE Digital Library<br>(IEEE) | 107                                             | 24                                               | 7                                                       | 7                                                       |
| ISI Web of Science<br>(WOS)    | 61                                              | 9                                                | 2                                                       | 2                                                       |
| Springer Link (SP)             | 1015                                            | 33                                               | 3                                                       | 3                                                       |

| Fuente              | Estudios<br>iniciales<br>(Primera<br>actividad) | Estudios<br>relevantes<br>(Segunda<br>actividad) | Estudios<br>primarios Tarea<br>1 (Tercera<br>actividad) | Estudios<br>primarios-Tarea<br>2 (Tercera<br>actividad) |
|---------------------|-------------------------------------------------|--------------------------------------------------|---------------------------------------------------------|---------------------------------------------------------|
| Science Direct (SD) | 53                                              | 1                                                | 0                                                       | 0                                                       |
| Scopus (SC)         | 332                                             | 13                                               | 4                                                       | 4                                                       |
| Google Scholar (GS) | 0                                               | 0                                                | 0                                                       | 3                                                       |
| Total               | 2032                                            | 92                                               | 20                                                      | 23                                                      |

Tabla 2 – Distribución de estudios por fuentes

Posteriormente, en la segunda actividad se aplicaron los criterios de exclusión (criterios de I&E), leyendo el título, el resumen y las palabras clave. Se encontraron y excluyeron 63 estudios duplicados. Después de aplicar los criterios de I&E, se excluyeron 1,877 estudios adicionales por no ser relevantes para responder las preguntas de investigación. Un total de 92 publicaciones se consideraron relevantes. Todos los resultados se muestran en el conjunto de datos del proceso de RSL (consultar <https://short.upm.es/xljfp> para obtener más información).

En la tercera actividad, se seleccionaron los estudios basándose en lo siguiente:

- Tarea 1, leyendo el texto completo. Si el estudio proporcionaba suficiente información, entonces se seleccionaba y se guardaba.
- Tarea 2, se aplicó una búsqueda adicional utilizando la técnica de bola de nieve descrita por C. Wohlin (2014). Como resultado, se agregaron dos estudios adicionales.

Después de aplicar la tercera actividad, se rechazaron 72 estudios y se aceptaron tres estudios de navegadores de Internet, obteniendo así 23 estudios aceptados después de la tercera actividad.

**2.3. Extraer y discutir los resultados de la revisión sistemática de la literatura**

Tras analizar los estudios primarios, se identificó que podrían clasificarse según los objetivos que debe cumplir la auditoría de riesgos de ciberseguridad. Para establecer la relación entre estos objetivos, resulta conveniente clasificar los conceptos y objetivos en cuatro objetivos principales, tomando como referencia el trabajo de Duncan et al. (2014).

Este trabajo encontró que los cuatro objetivos de la auditoría de riesgos de ciberseguridad se encuentran estrechamente relacionados, debido a que el proceso de auditoría de riesgos de ciberseguridad debe mejorarse además de realizar una evaluación, medir el cumplimiento y validar la garantía.

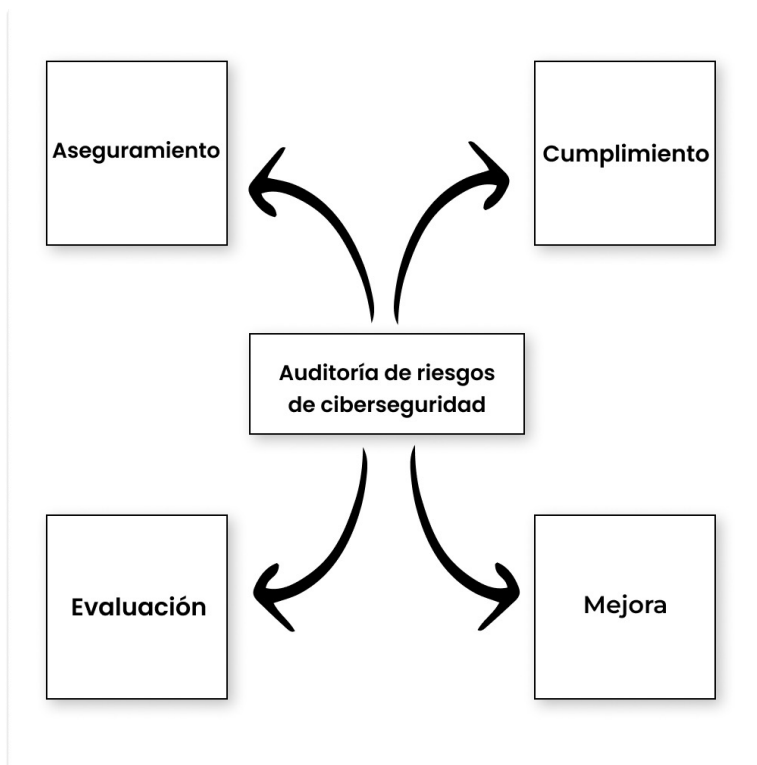


Figura 1 – Objetivos de auditoría de riesgos de ciberseguridad

Tras analizar los cuatro objetivos de la auditoría de riesgos de ciberseguridad, se determinó que se encuentran interrelacionados (ver Figura 1). Por ejemplo, uno de los objetivos de una auditoría es realizar una revisión para garantizar la efectividad de un sistema de gestión. Además, según el Diccionario Oxford de inglés, una auditoría es “un examen oficial de la calidad o estándar de algo”, por lo que una auditoría puede estar relacionada tanto con la garantía como con el cumplimiento de un estándar o regulación. El objetivo final de una auditoría de un sistema de gestión de ciberseguridad es mejorar el estado actual de la ciberseguridad.

Resumiendo, los resultados de esta sección, dentro de los conceptos relacionados con la ciberseguridad centrados en la realización de auditorías de riesgos, se identificó que los autores mencionan los objetivos “Evaluación”, “Garantía”, “Cumplimiento” y “Mejorar la Ciberseguridad”. Estos objetivos están interrelacionados, ya que constituyen los pilares fundamentales de las auditorías de riesgos de ciberseguridad.

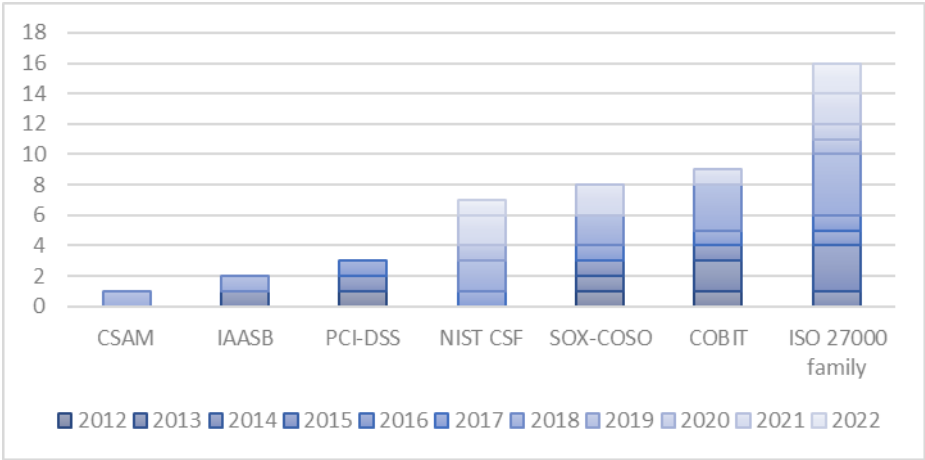


Figura 2 – Frecuencia de las guías de Auditoría de Ciberseguridad identificadas por año

Adicionalmente, se identificaron las siete guías utilizadas con mayor amplitud para las auditorías de riesgos de ciberseguridad: 1) la familia ISO 27000, 2) COBIT, 3) SOX-COSO, 4) NIST CSF, 5) PCI-DSS, 6) IAASB y 7) CSAM. De las guías previamente identificadas, solo el NIST CSF y el CSAM fueron creadas explícitamente para el ámbito de la ciberseguridad. Además, según los autores, el enfoque de las guías de auditoría de riesgos de ciberseguridad puede clasificarse en aquellas que se centran en procesos, tecnología o a nivel estratégico. (Véase Figura 2).

Una observación interesante es que se han identificado dos tendencias distintas en la investigación primaria. En primer lugar, algunos autores utilizan guías de auditoría centradas en el nivel estratégico sin considerar el componente técnico, y por otro lado, otros autores emplean guías de auditoría con un enfoque técnico sin considerar el enfoque estratégico. Estas dos tendencias demuestran la necesidad de complementarse mutuamente de manera equitativa para evaluar el sistema de gestión de ciberseguridad de manera objetiva y práctica.

### 3. Guía de riesgos de ciberseguridad propuesta

La guía Cybersecurity Risk Audit Guideline (CRAG) fue desarrollada considerando estándares internacionales tanto para ciberseguridad como para control interno. El propósito de la guía CRAG es abordar las auditorías de riesgos de ciberseguridad, teniendo en cuenta aspectos más allá de los técnicos y tradicionales.

La creación de la guía CRAG utilizó SADT como punto de partida, incorporando las fases de auditoría de control interno definidas en COSO ERM y complementándolas con las etapas de gestión de riesgos de seguridad propuestas por ISO 27001, adicionalmente se utilizó de referencia los modelos CSAM y NIST que son los que fueron identificados en la SLR (sección 2) como los utilizados para auditorías de riesgos de ciberseguridad.

### 3.1. SADT aplicado para crear la guía CRAG

Para la creación de la guía CRAG, se consideraron las siguientes fases (figura3), teniendo en cuenta y utilizando SADT:

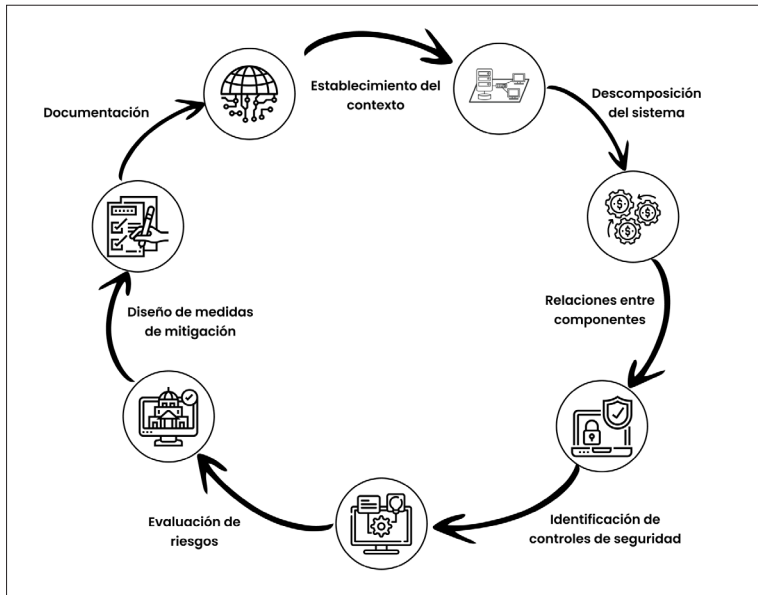


Figura 3 – Fases de la guía de auditoría CRAG

### 3.2.SADT Model Applied to Create CRAG Guideline

Para la creación de la guía CRAG, se consideraron las siguientes fases, teniendo en cuenta y utilizando SADT:

**3.2.1. Establecimiento del contexto:** Identificar el contexto de la auditoría de riesgos de ciberseguridad, considerando estándares y regulaciones aplicables. Dentro de esta fase de consideran las siguientes actividades:

- Establecer el alcance de la auditoría: Se definen los límites y la extensión de la auditoría.
- Identificar los objetivos de la auditoría: Se establecen los propósitos y objetivos de la auditoría.
- Identificar estándares y marcos de referencia: Se identifican los estándares y marcos que se utilizarán como base para evaluar los controles de ciberseguridad.
- Seleccionar una metodología de auditoría: Se define el enfoque y los métodos que se utilizarán durante la auditoría.
- Definir un equipo de auditoría: Se establece quiénes serían los miembros del equipo de auditoría.

**3.2.2. Descomposición del sistema:** Desglosar el sistema de ciberseguridad en sus principales componentes, como infraestructura tecnológica, activos de información, procesos comerciales y políticas de seguridad. Identificar áreas clave que deben evaluarse durante la auditoría. Estas áreas se definirán en función de clasificaciones generales de estándares como ISO 27001, NIST CSF (Ver Figura 3).

- Identificar la infraestructura tecnológica: Este componente se refiere al hardware, software e infraestructura de red que respalda los sistemas de información de una organización.
- Identificar los activos de información: Los activos de información abarcan los datos e información que una organización recopila, procesa, almacena y transmite.
- Identificar los procesos comerciales: Los procesos comerciales representan los flujos de trabajo, procedimientos y actividades que una organización lleva a cabo para alcanzar sus objetivos.
- Identificar las políticas de seguridad: Las políticas de seguridad establecen las reglas, pautas y mejores prácticas que rigen el enfoque de una organización hacia la ciberseguridad.
- Identificar las contramedidas de seguridad: Los controles de seguridad son las medidas técnicas y procedimentales implementadas para proteger los sistemas y datos de la organización.
- Identificar el plan de respuesta ante incidentes: Un plan de respuesta ante incidentes describe los pasos a seguir en caso de un incidente de ciberseguridad.
- Identificar los programas de concientización y entrenamiento en seguridad: El elemento humano es un componente crítico de un sistema de ciberseguridad.

**3.2.3. Relaciones entre componentes:** Determinar las interacciones y relaciones entre los componentes del sistema de ciberseguridad. Por ejemplo, cómo los activos de información están protegidos por controles de seguridad física definidos en políticas y procedimientos (Dyba et al., 2007) (Ver Figura 3).

- Identificar las relaciones entre los componentes previamente identificados: Las relaciones se establecen cuando existen dependencias entre más de un componente que necesita ser asegurado.
- Clasificar las relaciones según su criticidad: Establecer una calificación de criticidad considerando los posibles impactos si el proceso crítico respaldado por los componentes evaluados deja de funcionar o experimenta una falla de seguridad.

**3.2.4. Identificación de Controles de Seguridad:** Establecer cómo diferentes controles se relacionan con los componentes del sistema y contribuyen a mitigar los riesgos de ciberseguridad.

- Identificar al responsable de la contramedida: Definir a la persona responsable de la gestión y ejecución del control.
- Definir la Frecuencia de la Contramedida: Establecer cuántas veces se ejecuta el control dentro de un período de tiempo determinado.
- Identificar Dependencias con la Contramedida: Identificar si el control tiene

alguna dependencia con la ejecución de un proceso o control.

- Definir la Granularidad: Aclarar si el control opera de manera uniforme en toda la organización o si existen variaciones en su ejecución.
- Mantener la Integridad y Precisión: En el caso de la información utilizada para la ejecución del control (informes y registros), se debe considerar la integridad y precisión de los informes.

**3.2.5. Evaluación de Riesgos:** Evaluar los riesgos de ciberseguridad asociados con cada componente del sistema, considerando los controles de seguridad identificados y los requisitos de los estándares aplicables. Determinar el impacto y la probabilidad de ocurrencia de los riesgos identificados (Ver Figura 3).

- Recopilar muestras: Las muestras se utilizan para evaluar y obtener evidencia sobre una población o conjunto de datos más grande.
- Evaluar la efectividad operativa: La efectividad operativa de una contramedida se refiere a su capacidad para cumplir su propósito y lograr resultados deseados en el ámbito de la seguridad.
- Identificar desviaciones: El proceso de detectar discrepancias o irregularidades entre la información examinada y los estándares, políticas, leyes o regulaciones aplicables.
- Identificar observaciones: Las observaciones de control son hallazgos o resultados identificados durante una auditoría que indican deficiencias en los controles internos de una organización.
- Evaluar la probabilidad de fallo de la contramedida: La probabilidad de fallo de una contramedida se refiere a la posibilidad de que una contramedida no funcione como se espera o no sea efectiva para mitigar o contrarrestar una amenaza o riesgo específico.
- Evaluar el impacto de la materialización del riesgo: La evaluación del impacto del riesgo tiene como objetivo evaluar las posibles consecuencias o efectos negativos que podrían ocurrir si un riesgo se materializa.

**3.2.6. Diseño de Medidas de Mitigación:** Establecer acciones correctivas y preventivas para reducir los riesgos a un nivel aceptable (Ver Figura 3).

- Establecer planes de acción: El objetivo principal del plan de acción es abordar las deficiencias identificadas, implementar medidas correctivas y mejorar los procesos y controles internos de la organización.
- Monitorear planes de acción: El monitoreo de los planes de acción de la auditoría es crucial para garantizar que las acciones correctivas se implementen de manera efectiva y que las deficiencias identificadas se resuelvan.
- Finalizar planes de acción: El cierre del plan de acción de la auditoría se realiza una vez que todas las acciones correctivas han sido implementadas y las deficiencias identificadas en la auditoría han sido abordadas.

**3.2.7. Documentación:** Documentar los resultados de la auditoría de riesgos de ciberseguridad. Esto puede incluir la creación de diagramas que muestren la estructura del sistema de ciberseguridad, los riesgos identificados, los controles de seguridad recomendados y las medidas de mitigación propuestas (Ver Figura 3).

4. Validación de la guía CRAG

Para llevar a cabo esta sección, decidimos aplicar un caso de estudio, tomando como referencia el trabajo de Yin (2018), que proporciona pautas para la realización de casos de estudio. Utilizamos como objeto de estudio un proceso de ciberseguridad en un banco internacional. La estructura del banco objeto de estudio está compuesta por 115.000 empleados y más de 89 millones de clientes a nivel mundial. Por lo tanto, el caso de estudio se limitará a la sede corporativa de México, que cuenta con 10.001 empleados con sede en la Ciudad de México.

El estudio de caso se llevó a cabo desde el noviembre de 2023, hasta el 30 de marzo de 2024. Comenzó con la presentación y mención de que la aplicación de este caso de estudio tenía propósitos puramente académicos y se centró en uno de los procesos de un departamento en el campo de la ciberseguridad.

Debido a su importancia en las revisiones de estados financieros por auditores externos, auditores internos y reguladores, el proceso seleccionado estaba relacionado con el ciclo de vida del usuario y era gestionado por el departamento de Gestión de Identidad y Acceso (IAM, por sus siglas en inglés).

El self risk assessment consiste en una evaluación independiente del control interno que no es formalmente una auditoría con el fin de identificar la madurez de los controles. Para la aplicación del caso de uso se propuso realizar por el área de control interno un “self risk assessment” (International Organization for Standardization, 2018).

Para esta autoevaluación se consideró la metodología de evaluación de la primera y segunda línea de defensa de la organización incorporando las fases y actividades de CRAG. Es relevante aclarar que CRAG no tiene por intención sustituir ninguna metodología de auditorías o evaluaciones internas de los controles de riesgos de ciberseguridad, el objetivo de CRAG es servir de guía para robustecer y facilitar la ejecución de auditorías de riesgos de ciberseguridad.

|                                           |                                                                            | Formulario CRAG |
|-------------------------------------------|----------------------------------------------------------------------------|-----------------|
| Establecimiento del contexto              | Establecer el alcance de la auditoría:                                     |                 |
|                                           | Identificar los objetivos de la auditoría:                                 |                 |
|                                           | Identificar estándares y marcos de referencia:                             |                 |
|                                           | Seleccionar una metodología de auditoría:                                  |                 |
| Descomposición del sistema                | Definir un equipo de auditoría:                                            |                 |
|                                           | Identificar la infraestructura tecnológica:                                |                 |
|                                           | Identificar los activos de información:                                    |                 |
|                                           | Identificar los procesos comerciales:                                      |                 |
|                                           | Identificar las políticas de seguridad                                     |                 |
|                                           | Identificar las contramedidas de seguridad                                 |                 |
|                                           | Identificar el plan de respuesta ante incidentes                           |                 |
|                                           | Identificar los programas de concientización y entrenamiento en seguridad: |                 |
| Relaciones entre componentes:             | Identificar las relaciones entre los componentes previamente identificados |                 |
|                                           | Clasificar las relaciones según su criticidad:                             |                 |
| Identificación de Controles de Seguridad: | Identificar al responsable de la contramedida                              |                 |
|                                           | Definir la Frecuencia de la Contramedida                                   |                 |
|                                           | Identificar Dependencias con la Contramedida                               |                 |
|                                           | Definir la Granularidad:                                                   |                 |
| Mantener la Integridad y Precisión:       |                                                                            |                 |

Figura 4 – Formulario fases y actividades CRAG

Para ello, la oficina del chief information security officer (CISO) llenó un formulario (disponible en el siguiente enlace: <https://short.upm.es/xb9e4>) relacionado cada fase y

sus actividades ver figura 4. Este formato sirve como evidencia del informe de auditoría y resumen ejecutivo de la auditoría. El objetivo del formato es validar la correcta diferenciación de cada una de las fases y la aplicación de cada una de las actividades.

Una vez llenado el formulario (formulario completado disponible en el siguiente enlace: <https://short.upm.es/r7gko>) se identificaron de un total de 20 aplicaciones en el alcance un total de 30 controles con un total de 3 fallos o desviaciones en el diseño del control (monitorización y gestión de cuentas privilegiadas de plataformas) y un total de 17 fallas en la eficacia operativa del control (bajas, bloqueos certificación y segregación de funciones de usuarios/perfiles).

Posteriormente de la implementación de la guía de auditoría CRAG el CISO identificó dentro de los tiempos invertidos para el self assessment un incremento de 1 semana a 2 semanas en la planeación del risk assessment por la utilización de la guía CRAG. A pesar del incremento en el tiempo de planeación el CISO validó que la ejecución del self assessment fue más practica por la incorporación de actividades definidas y que este se alineaba con los pasos utilizados en la metodología tradicional utilizada por la institución en las revisiones de control interno.

## 5. Conclusiones

En este se estudio ha demostrado que los autores de los estudios referenciados mencionan a la auditoría de ciberseguridad sin tomar en consideración los conceptos asociados ni los objetivos mínimos. Además, la auditoría de ciberseguridad está aún en sus inicios y le queda camino por recorrer antes de alcanzar la madurez necesaria. Otro punto mencionado en los estudios referidos indica que, dado las auditorías de TI, y especialmente las de ciberseguridad, no están tan estrictamente supervisadas como las auditorías contables, se cuestiona su efectividad.

Este estudio ha encontrado que en general, el cumplimiento, el aseguramiento y la auditoría son conceptos diferentes dentro de la ciberseguridad. Debido al costo que implica contratar especialistas en estos temas, los responsables de la aplicación de las directrices de auditoría intentan controlar el problema, limitándose a un ejercicio de marcar casillas. Siendo una actividad que no engloba el escenario real de una auditoría.

En 17 de los 23 estudios de auditoría analizados, los objetivos definidos que conforman la auditoría de riesgos de ciberseguridad: “Evaluar”, “Aseguramiento”, “Cumplimiento” y “Mejora de la ciberseguridad”. Como resultado de la relación identificada entre los objetivos, se considera que podría desarrollarse una nueva línea de investigación para establecer la viabilidad de desarrollar una guía de auditoría que cubra estos temas.

Actualmente, hay pocas directrices de auditoría enfocadas específicamente en los requisitos de ciberseguridad. Los modelos CSF (Cybersecurity Framework) y CSAM (Cybersecurity Assessment and Management) del NIST han sido los únicos identificados en esta investigación como directrices para auditar la madurez y eficacia de los mecanismos de mitigación de riesgos de ciberseguridad.

Adicionalmente se realizó una propuesta tomando como referencia las guías identificadas en la SLR, la cual se denominó Cybersecurity Risk Audit Guideline (CRAG). Esta guía

para realización de auditorías de riesgos de ciberseguridad se desarrolló considerando las 3 fases de la auditoría definidas por lo estudios primarios de la SLR 1) Establecimiento del contexto, 2) Realización de la auditoría y 3) preparación del informe de auditoría.

Para esto SDTA fue utilizado para obtener un total de 7 fases que deben ser consideradas en la auditoría de riesgos de ciberseguridad. De las cuales se desprenden un total de 28 actividades.

Igualmente, al desarrollar un caso de estudio se diseñó un formulario que sirva de punto inicial como una guía donde se documente la auditoría de riesgos de ciberseguridad. El objetivo del formulario es obtener un punto de partida para recoger opiniones y comentarios de diferentes organizaciones para poder consolidar una guía eficiente para la realización de riesgos de ciberseguridad.

### **5.1. Limitaciones del presente trabajo de investigación**

Una limitación o sesgo potencial es la dificultad para comparar y obtener puntos de referencia comunes para las directrices que tienen enfoques tan diferentes, como las directrices de control interno, las directrices de seguridad de la información y las directrices de ciberseguridad.

Se recomienda la guía CRAG para evaluar contramedidas específicas de ciberseguridad, ya que al aplicarla a contramedidas creadas para otra área posible, podría no tener en cuenta las necesidades específicas de esta área (por ejemplo, riesgos de calidad, etc.).

Una de las limitaciones del presente trabajo es realizar a una mayor escala la validación del caso de estudio de CRAG para la identificación de mejoras y validación correcta de la eficiencia de esta guía.

Otra limitación puede ser la omisión de algún parámetro o características relevantes consideradas en las guías tomadas como referencia, para la creación de CRAG. Para ello se propone realizar un modelo de comparación y análisis extendido de todas las características recogidas en las guías para auditorías de riesgos de ciberseguridad.

Se propone como trabajo futuro realizar una comparativa y extracción de parámetros, conceptos y características de las guías para auditorías de riesgos de ciberseguridad. Adicionalmente relacionar los comentarios e información en formatos llenados por los CISO's de diferentes organizaciones y utilizar esta información para identificar patrones o relaciones entre actividades, conceptos y características para poder establecer algún parámetro cuantitativo que ayude en la ejecución de las auditorías de riesgos de ciberseguridad.

## **Referencias**

Congram, C., & Epelman, M. (1995). How to describe your service. *International Journal of Service Industry Management*, 6(2), 6–23. <https://doi.org/10.1108/09564239510084914>

- Duncan, B., & Whittington, M. (2014). Compliance with standards, assurance and audit: Does this equal security? ACM International Conference Proceeding Series, 2014-September, 77–84. <https://doi.org/10.1145/2659651.2659711>
- Dyba, T., Dingsoyr, T., & Hanssen, G. K. (2007). Applying Systematic Reviews to Diverse Study Types: An Experience Report. First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007), 225–234. <https://doi.org/10.1109/ESEM.2007.59>
- European Confederation of Institutes of Internal Auditors. (2020). “Risk in focus 2021. Hot topics for internal auditors.” Available at: <https://www.eciia.eu/Wp-Content/Uploads/2020/09/100242-RISK-IN-FOCUS-2021-52PP-ECIIA-Online-V2.Pdf>.
- Ezzamouri, N., & Hulstijn, J. (2018). Continuous monitoring and auditing in municipalities. Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age, 1–10. <https://doi.org/10.1145/3209281.3209301>
- Fernandez, A., Black, J., Jones, M., Wilson, L., Salvador-Carulla, L., Astell-Burt, T., & Black, D. (2015). Flooding and mental health: A systematic mapping review. In PLoS ONE (Vol. 10, Issue 4). Public Library of Science. <https://doi.org/10.1371/journal.pone.0119929>
- Gale, M., Bongiovanni, I., & Slapnicar, S. (2022). Governing cybersecurity from the boardroom: Challenges, drivers, and ways ahead. Computers & Security, 121, 102840. <https://doi.org/10.1016/j.cose.2022.102840>
- Galligan, M. E., & Rau, K. (2015). COSO in the cyber age.
- Gauthier, M. P., & Brender, N. (2021). How do the current auditing standards fit the emergent use of blockchain? Managerial Auditing Journal, 36(3), 365–385. <https://doi.org/10.1108/MAJ-12-2019-2513>
- Ibrahim, A., Valli, C., McAteer, I., & Chaudhry, J. (2018). A security review of local government using NIST CSF: a case study. Journal of Supercomputing, 74(10), 5171–5186. <https://doi.org/10.1007/s11227-018-2479-2>
- Islam, Md. S., Farah, N., & Stafford, T. F. (2018). Factors associated with security/cybersecurity audit by internal audit function. Managerial Auditing Journal, 33(4), 377–409. <https://doi.org/10.1108/MAJ-07-2017-1595>
- Information Systems Audit and Control Association. (2018). COBIT 2019. [www.isaca.org/COBIT](http://www.isaca.org/COBIT)
- International Organization for Standardization. (2018). ISO IEC 27000 2018 Information technology - Information security Management systems - Overview and vocabulary (pp. 1–26).
- Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering - A systematic literature review. In Information and Software Technology (Vol. 51, Issue 1, pp. 7–15). Elsevier B.V. <https://doi.org/10.1016/j.infsof.2008.09.009>

- National Institute of Standards and Technology. (2018). NIST cybersecurity framework. Proceedings of the Annual ISA Analysis Division Symposium, 535, 9-25.
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 1–18. <https://doi.org/10.1016/j.infsof.2015.03.007>
- Sabillon, R., Serra-Ruiz, J., Cavaller, V., & Cano, J. (2018). A comprehensive cybersecurity audit model to improve cybersecurity assurance: The cybersecurity audit model (CSAM). In *Proceedings - 2017 International Conference on Information Systems and Computer Science, INCISCOS 2017* (pp. 253–259). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/INCISCOS.2017.20>
- Slapničar, S., Vuko, T., Čular, M., & Drašček, M. (2022). Effectiveness of cybersecurity audit. *International Journal of Accounting Information Systems*, 44, 100548. <https://doi.org/10.1016/j.accinf.2021.100548>
- Stafford, T., Deitz, G., & Li, Y. (2018). The role of internal audit and user training in information security policy compliance. *Managerial Auditing Journal*, 33(4), 410–424. <https://doi.org/10.1108/MAJ-07-2017-1596>
- Steinbart, P. J., Raschke, R. L., Gal, G., & Dilla, W. N. (2012). The relationship between internal audit and information security: An exploratory investigation. *International Journal of Accounting Information Systems*, 13(3), 228-243. <https://doi.org/10.1016/j.accinf.2012.06.007>
- Steinbart, P. J., Raschke, R. L., Gal, G., & Dilla, W. N. (2016). SECURQUAL: An Instrument for Evaluating the Effectiveness of Enterprise Information Security Programs. *Journal of Information Systems*, 30(1), 71–92. <https://doi.org/10.2308/isis-51257>
- Yin, R. K. (2018). *Case Study Research and Applications (Sixth Edition)*. SAGE Publications India Pvt. Ltd. <https://lcn.loc.gov/2017040835>



# Modelos de identificación de enfermedades cardiovasculares implementando técnicas de aprendizaje máquina: una revisión sistemática de la literatura

Johan Mardini - Bovea<sup>1</sup>, Dixon Salcedo<sup>1</sup>, Emiro De-la-Hoz-Franco<sup>1</sup>, Yadira Quiñonez<sup>2</sup>, Luz Jimenez-Roa<sup>3</sup>, Víctor De la Hoz-Avila<sup>1</sup>, Jhon González Sanchez<sup>1</sup>

[jmardini@cuc.edu.co](mailto:jardini@cuc.edu.co); [dsalcedo2@cuc.edu.co](mailto:dsalcedo2@cuc.edu.co); [edelahoz@cuc.edu.co](mailto:edelahoz@cuc.edu.co); [yadiraqui@uas.edu.mx](mailto:yadiraqui@uas.edu.mx); [lijimenezr@sena.edu.co](mailto:lijimenezr@sena.edu.co); [vdelahoz10@cuc.edu.co](mailto:vdelahoz10@cuc.edu.co); [jgonzale120@cuc.edu.co](mailto:jgonzale120@cuc.edu.co)

<sup>1</sup> Departamento de Ciencias de la Computación y Electrónica, Universidad de la Costa–CUC, Barranquilla 080002, Colombia.

<sup>2</sup> Facultad de informática de Mazatlán-FIMAZ, Universidad Autónoma de Sinaloa, Mazatlán 82000, México.

<sup>3</sup> Centro de Enseñanza Colombo Alemán, SENA Barranquilla 08004, Colombia.

DOI: 10.17013/risti.53.87–105

**Resumen:** El uso de técnicas de Aprendizaje Automático (AA) en el área de la salud, específicamente en la identificación de enfermedades cardiovasculares (IEC), ha tenido un impacto significativo debido a la capacidad para analizar grandes cantidades de datos y extraer información relevante que puede ser esencial para la toma de decisiones médicas. Sin embargo, antes de ponerlos a disposición de los usuarios finales (médicos), se debe evaluar su capacidad para detectar sintomatologías relacionadas con enfermedades cardíacas utilizando puntos de referencia de conjuntos de datos en escenarios experimentales. Por lo tanto, es complicado determinar qué características utilizar en el proceso de evaluación y qué técnicas de ML son más adecuadas para la predicción de IEC. Este artículo presenta una revisión sistemática de la literatura sobre el procesamiento de conjuntos de datos basados en pruebas clínicas de enfermedades cardiovasculares y técnicas de AA. En este sentido, se realizó un análisis de las diferentes variables extraídas de publicaciones de revistas indexadas en bases de datos especializadas como Scopus, Web of Science, Science Direct, Biomed y Pubmed.

**Palabras-clave:** Enfermedades cardiovasculares, técnicas de inteligencia artificial, conjunto de datos, síndrome coronario agudo.

***Models of identification cardiovascular diseases implementing machine learning techniques: a systematic literature review***

**Abstract:** The use of Machine Learning (ML) techniques in the health area, specifically in the identification of cardiovascular diseases (IEC), has had a significant impact due to the ability to analyze large amounts of data and extract

relevant information that can be essential for medical decision-making. However, before making them available to end users (doctors), their ability to detect heart disease-related symptomatology should be evaluated using benchmark data sets in experimental settings. Therefore, determining which features to use in the evaluation process and which ML techniques are most suitable for IEC prediction is complicated. This article presents a systematic literature review on processing cardiovascular disease clinical trial-based datasets and ML techniques. In this sense, the different variables were analyzed from journal publications indexed in specialized databases such as Scopus, Web of Science, Science Direct, Biomed, and PubMed.

**Keywords:** Cardiovascular diseases, artificial intelligence techniques, data set, acute coronary syndrome.

## 1. Introducción

La cardiopatía isquémica es responsable del 42% de las muertes por enfermedades cardiovasculares a nivel mundial, mientras que las enfermedades cerebrovasculares son responsables del 34%. Se estima que 17,7 millones de personas murieron como resultado de esta causa en 2015, lo que representa el 31% de todas las muertes registradas en todo el mundo (Organización Panamericana de la Salud, s. f.). Es decir, 7,4 millones de estas muertes fueron causadas por enfermedades cardíacas. Más de tres cuartas partes de las defunciones por ECV se producen en los países de ingresos bajos y medios. De los 17 millones de muertes de personas menores de 70 años causadas por enfermedades no transmisibles, el 82% se registra en países de ingresos bajos y medios, y el 37% se debe a enfermedades cardiovasculares (ECV), (Organización Mundial de la Salud, s. f.). De acuerdo con informes de la Organización Mundial de la Salud (OMS) (Organización Mundial de la Salud, s. f.) y la Organización Panamericana de la Salud (OPS) (Instituto Nacional de Salud, s. f.), las enfermedades cardiovasculares, junto con el cáncer, la diabetes y las enfermedades pulmonares crónicas, se identifican como enfermedades que están aumentando rápidamente y se clasifican como las principales causas de muerte en el mundo. Las enfermedades cardiovasculares son la causa más importante de muerte, superando en gran medida el cáncer, la diabetes mellitus y las enfermedades pulmonares crónicas.

En la actualidad, las instituciones de salud están demandando cada vez más la incorporación de la tecnología. Los sistemas de IEC identifican enfermedades cardíacas, lo que mejora la calidad de vida de los pacientes al permitir una identificación temprana de ECV. El área de investigación de ECV ha tenido un impacto significativo en la creación de soluciones pre-evento cardíaco, como el apoyo a los pacientes con SCA. La aplicación de múltiples soluciones en entornos clínicos monitoreados con datos de exámenes médicos demuestra esto (Yao, 2020). Basados en pruebas relacionadas con la enfermedad cardiovascular, los objetivos de los sistemas de IEC son los siguientes: 1) la creación de modelos predictivos que permitan identificar de forma bi-clase la tendencia a padecer enfermedad cardiovascular; 2) proporcionar al equipo médico las herramientas necesarias basadas en software para identificar alguna enfermedad cardiovascular; y 3) estratificar el riesgo de los pacientes mediante la combinación de puntuaciones de riesgo validadas.

Los datos recopilados de sensores en entornos controlados se almacenan en un conjunto de datos incluyendo pruebas de laboratorio, exámenes diagnóstico y pruebas de esfuerzo clínicas. Se han sugerido varias formas de recopilar datos, como biomarcadores (Idris, 2020), pruebas clínicas (Narayan, 2019), sensores biomédicos (Metsker, 2020; Mendes, 2018; González-Alcaide, 2020) y pruebas de riesgo cardiovascular (Xue, 2020; Rezaianzade, 2020; Brandberg, 2020; Konys, 2019). Luego, el conjunto de datos se utiliza para entrenar varias técnicas de inteligencia artificial, incluido el aprendizaje automático, con el objetivo de predecir las tendencias de los pacientes hacia una enfermedad específica de SCA. Esto daría una alerta al equipo médico adecuado y reduciría los riesgos relacionados con el deterioro de la salud de los pacientes analizados. Actualmente existen conjuntos de datos para ECV. En consecuencia, determinar qué características o evaluaciones médicas utilizar en el proceso de evaluación de un sistema de IEC y qué técnicas implementar en las fases de construcción del modelo (preprocesamiento, selección de características, clasificación y postclasificación) son las más adecuadas para mejorar los índices de identificación de algunas enfermedades relacionadas con SCA.

Los sistemas IEC se derivan de los avances tecnológicos en los sensores biomédicos y en los resultados de pruebas médicas en tiempos extremadamente cortos, o incluso instantáneos. Como se detalla en (Chu, 2020; Magoev, 2018; Metsker, 2017; Si, 2021), hay muchas aplicaciones de IEC, pero se destacan la detección de riesgo cardiovascular (Kutyrev, 2019), la detección del Síndrome Coronario Agudo con Elevación del Segmento ST (SCACEST) (Xie, 2020), la identificación de angina inestable (AI) (Sun, 2020) y el Síndrome Coronario Agudo sin Elevación del Segmento ST (SCASEST) (Iannatton, 2020), entre otros. Por lo tanto, es extremadamente difícil evaluar la confiabilidad de los IEC en términos de su capacidad para predecir una variedad de enfermedades cardíacas, especialmente las del SCA recopiladas en el conjunto de datos. Con el desarrollo de nuevas IEC, la recolección de datos se ha vuelto más difícil. Esto se debe a que la recolección de datos registrados en Historias Clínicas Electrónicas (HCE) o Registro Médico Electrónico (RME) no incluye una variedad de pruebas médicas y exámenes diagnósticos para evaluar los IEC con suficiente rigor. Una variedad de puntos de referencia del conjunto de datos ha surgido como resultado de esta situación. Estos incluyen el registro de Coronary Heart Attack Ireland Register (CHAIR), Global Registry of Acute Coronary Event (GRACE), Myocardial Infarction National Audit Project (MINAP), Spanish Register form Non-ST Elevation MI (DESCARTES), Acute Coronary Syndrome Registry International (ACOSRI), entre otros.

Teniendo en cuenta lo anterior, este trabajo tiene como objetivo que los investigadores desarrollen IEC y los coloquen a prueba en el contexto de pruebas de laboratorio, para enviar sus propuestas a evaluación utilizando distintos puntos de referencia de conjunto de datos, y finalmente, validar sus desarrollos con una muestra con pacientes y así contrastar efectividad de los mismos. La aplicación de tecnologías emergentes como las técnicas de AA, que todavía están en proceso de investigación y evolución, es indudable que el sector de la salud es uno de los muchos contextos en los que se pueden utilizar, porque aún existe un largo camino por recorrer en este ámbito de aplicación, y el uso de estas herramientas presenta múltiples retos y desafíos. Los siguientes son algunos de los principales hallazgos que este documento ha hecho para el campo de estudio:

1. Encontrar modelos de enfermedades cardiovasculares basados en AA.
2. Encontrar modelos basados en AA para encontrar enfermedades de SCA.
3. Identificación y evaluación del riesgo cardiovascular, incluidas sus aplicaciones y avances.

Debido a que no existe un punto de referencia que haya identificado los conjuntos de datos más relevantes en términos de los sistemas de IEC, el enfoque propuesto en este documento es una contribución original. Este artículo proporciona una caracterización detallada y posterior análisis de las diferentes técnicas de clasificación utilizadas para enfermedades cardiovasculares, incluidas las técnicas de distribución del conjunto de datos utilizado para entrenamiento y pruebas, así como las métricas de calidad (precisión, exactitud) utilizadas en la evaluación de IEC. Un volumen considerable de artículos a revisar se compone de la condensación, documentación y análisis de las variables mencionadas para cada conjunto de datos y estudios realizados entre 2019 a 2024.

Todas estas propuestas se basan en herramientas basadas en las Tecnologías de la Información y las Comunicaciones (TIC), como el Internet de las cosas (IoT), los sistemas independientes (para la captación de datos y la gestión de los procesos de comunicación entre sensores y repositorios de datos), las plataformas de computación en la nube (para el almacenamiento y procesamiento en la nube de grandes volúmenes de datos capturados en tiempo real), y la construcción de modelos predictivos basados en inteligencia artificial. Específicamente, utilizando técnicas de AA (para encontrar relaciones no triviales entre varias variables cardiovasculares y predecir comportamientos futuros a partir de la información histórica de los pacientes).

Este artículo científico está organizado de la siguiente manera: la investigación relacionada se describe en la sección 2. La metodología para realizar una revisión exhaustiva de la literatura se presenta en la Sección 3. El análisis cuantitativo se encuentra en la Sección 4. Los análisis técnicos se encuentran en la sección 5, se incluye una caracterización de cada conjunto de datos identificado, así como análisis de resultados correspondientes. Finalmente, la conclusión y el trabajo futuro se presentan en la sección 6.

## 2. Investigación relacionada

Al revisar literatura relacionada, se ha observado que algunas revisiones de la literatura analizan enfoques tecnológicamente muy específicos (técnicas de AA e hibridación (Manterola, 2013; García-Pérez, 2013), estudios comparativos de técnicas de riesgo cardiovascular (Merlano-Porras, 2013), (Henriquez, 2022), sistema de monitoreo cardiovascular (Kitchenham, 2009)). Aunque no son completas, otras revisiones se limitan a evaluar la precisión de las técnicas de clasificación para predecir enfermedades cardiovasculares a partir de conjuntos de datos (Grams, 2011; Comas-Gonzalez, 2023; Sanchez, 2016). Sin embargo, no hay pruebas que sugieran una revisión exhaustiva de la literatura que permita caracterizar conjuntos de datos de ECV enfocados en el SCA. La siguiente es una descripción detallada de los estudios mencionados anteriormente:

Kitchenham et al. (Kitchenham, 2009), usan técnicas asistidas por computadora para proporcionar herramientas rápidas y precisas para identificar las señales de

electrocardiograma (ECG) de un paciente. También resumen y comparan métodos de diagnóstico utilizando preprocesamiento de datos, ingeniería de características, clasificación y aplicación. Además, se lleva a cabo un modelo extremo a extremo que integra la extracción y clasificación de características en algoritmos de aprendizaje en (Manterola, 2013). Esto hace que el proceso de análisis de datos sea mucho más sencillo y muestra una precisión evaluada de alta calidad. En (García-Pérez, 2013), los autores examinaron dispositivos portátiles que permiten a los usuarios monitorear su estado cardiovascular en cualquier momento. Estos dispositivos ofrecen nuevos escenarios y plantean desafíos para la aplicación de algoritmos para identificar enfermedades utilizando electrocardiogramas. De manera similar, en (Merlano-Porras, 2013) se implementó la extracción de características clínicas multivariadas de pacientes con SCA registrados en una base de datos, y se crearon varios modelos que predecían con éxito un paro cardíaco en pacientes con SCA utilizando diferentes algoritmos de AA.

En (Henriquez, 2022) proponen el desarrollo de un modelo predictivo basado en hibridación para la predicción de enfermedades cardiovasculares utilizando el conjunto de datos Cleveland Heart Disease mediante técnicas de selección de características estadísticas con aprendizaje automático y redes neuronales artificiales (SOM Y GHSON). Los autores en (Comas-Gonzalez, 2023) realizaron un estudio comparativo de métodos de inteligencia artificial no supervisados para encontrar el modelo de predicción cardiovascular más efectivo. En (Sanchez, 2016), se comparó la predicción del aprendizaje automático con modelos de predicción de riesgos cardiovascular existentes para pacientes con SCA, como Registry of Acute Coronary Events (GRACE), National Early Warning Score (NEWS) y Modified Early Warning Score (MEWS).

### 3. Metodología

La Revisión Sistemática de la Literatura (RSL) es una pieza clave de la investigación que permite establecer las bases para investigaciones futuras. Kitchenham et al (Kitchenham, 2009) proponen una metodología basada en la definición de preguntas de investigación, proceso de búsqueda, criterios de inclusión y exclusión, evaluación de la calidad, recolección de datos, análisis de datos y desviaciones del protocolo. La gran mayoría de las investigaciones de este tipo se realizan en el campo de la salud, para lo cual se analizaron las metodologías propuestas en (Manterola, 2013; García-Pérez, 2013; Comas-Gonzalez, 2023). En el campo de la ingeniería, los autores (Merlano-Porras, 2013; Henriquez, 2022; Comas-Gonzalez, 2023; Sanchez, 2016; Grams, 2011; Crimi, 2020) proponen sus respectivas metodologías. Para el desarrollo de esta RSL se basará en la implementación de (Knowles, 2018), lo cual se organiza en tres etapas: la definición de parámetros de búsqueda (objetivo, hipótesis e índice de búsqueda), identificación y depuración en bases de datos (selección de cadenas de búsqueda cuyos resultados se profundizarán) y la propuesta de respuestas a la hipótesis (de la información obtenida, categorización y análisis de los artículos más relevantes).

En particular, en (Kandasamy, 2018) se propone un método para validar la Enfermedad Renal Crónica (ERC) y otras afecciones relacionadas en conjuntos de datos existentes, incluidos los registros de enfermedades y los conjuntos de datos administrativos. Cada uno de estos estudios contribuyó al método RSL utilizado en la investigación, que se organizó en tres etapas. La primera etapa comienza con la determinación de los objetivos

de la revisión y consiste en identificar la siguiente hipótesis: “¿Qué conjunto de datos de SCA y técnicas de AA ha tenido un mayor impacto en el desarrollo de investigaciones asociadas a IEC?”. Posteriormente, se procedió a localizar los temas de las bases de datos en las que se centraría la búsqueda. Estas bases de datos incluyen Scopus, Web of Science, Science Direct, Biomed y Pubmed. Finalmente, se identificaron y validaron las palabras clave a utilizar y descartar debido al ruido que produce en los resultados.

El esquema utilizado para crear los términos de búsqueda se muestra en la Figura 1 (a). Se ha eliminado el término “imagen” porque el procesamiento de datos de este tipo de investigación es muy diferente al de otros mecanismos de retroalimentación, como biomarcadores, pruebas invasivas y no invasivas, así como exámenes diagnósticos cardiovasculares. Adicionalmente, se estableció una ventana de búsqueda de 2019 a 2024 para determinar el alcance de las publicaciones a analizar.

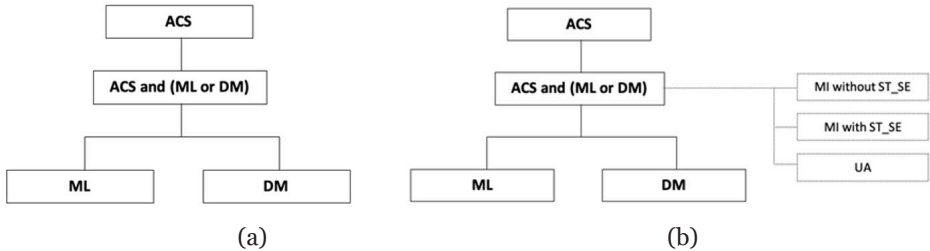


Figura 1 – (a) Cadena general de búsqueda (b) Cadena de específica de búsqueda

En la segunda etapa, se identificaron y filtraron datos de bases de datos especializadas y se analizaron los resultados de varias cadenas de búsqueda. Los datos se han representado sintéticamente en una variedad de arreglos y los índices de búsqueda se han organizado jerárquicamente, eliminando las combinaciones que no dieron ningún resultado. Además, se descubrieron los términos que produjeron una gran cantidad de resultados, y se agregaron criterios más precisos para limitar las búsquedas al tema de la investigación. Se seleccionaron las cadenas de búsqueda que produjeron resultados al identificar artículos que coinciden con la hipótesis sugerida. La Figura 1(b) muestra las palabras clave utilizadas para crear las cadenas de búsqueda, cuya estructura es la siguiente: (“Acute Coronary Syndrome”) OR (“Myocardial Infarction without ST-segment Elevation”) OR “Myocardial Infarction with ST-segment Elevation” OR “Angina Inestable”) AND (Machine Learning) AND NOT (“Image”).

Se registraron varias variables cuantitativas para cada artículo de investigación, que incluían el año de publicación, la revista, la tipología del documento, el cuartil de la revista, el país de publicación, el país de producción y la entidad o universidad que presentó el producto. Además, se registraron múltiples variables técnicas para describir el tipo de conjunto de datos utilizado en los varios trabajos de investigación consultados. Finalmente, los clasificadores a los que se hace referencia en los artículos, se incluye su enfoque, segmentación, representación, selección de características, equilibrio y adición de instancias, categoría y subcategoría. La última etapa consiste en presentar y analizar los resultados.

#### 4. Análisis bibliométrico

Después de realizar el análisis de las variables bibliométricas de las 245 publicaciones que se encuentran en bases de datos de revistas indexadas de alto impacto, se utilizaron diferentes métodos para medir las publicaciones. Estos métodos incluyeron el año de publicación, el número de artículos publicados en cada base de datos, la tipología de los artículos y el cuartil de la revista, congreso o libro en la que se publicaron. Además, se examinaron los países que reciben y producen un mayor flujo de trabajos, así como las revistas y universidades que han avanzado más en este campo de investigación.

La Figura 2 muestran una tendencia creciente en el número de publicaciones relacionadas con enfermedades cardiovasculares en términos de: análisis de síndrome coronario agudo, descripción general de enfermedades cardiovasculares, uso de tecnología para identificar cardiopatías y sistemas de evaluación de riesgo cardiovascular. La Figura 2 (a) muestra cómo el número de publicaciones en este campo de investigación ha aumentado cada año, alcanzando su punto máximo en 2022 con 71 documentos. La figura 2 (b) indica que Science Direct es la base de datos científica con el mayor número de productos de investigación registrados entre 2019 y 2023.

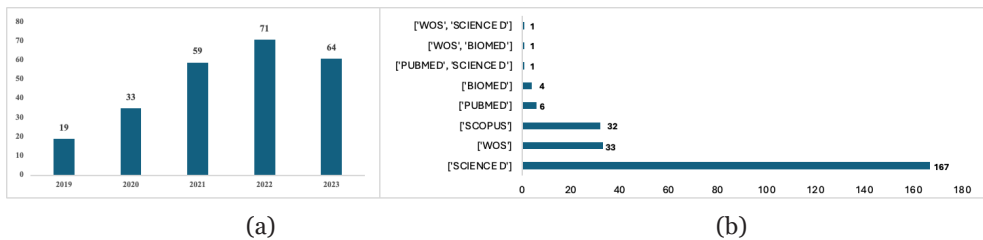


Figura 2 – (a) Cantidad de trabajos en el área de enfermedades cardiovasculares publicados en marco de tiempo, (b) Cantidad de trabajos en el área enfermedades cardiovasculares por

El 68,16% de las publicaciones en esta área de estudio se encuentran en revistas, las cuales se pueden acceder a través de la base de datos Science Direct, que es especializada en este tema. El 85% de las publicaciones accesibles desde diversas bases de datos especializadas son revistas, mientras que el 15 % son libros, capítulos de libro y actas. El mayor porcentaje de publicaciones se realiza en recursos categorizados en primer cuartil (55%), segundo cuartil (20%), tercer cuartil (8%), cuarto cuartil (1%) y productos no categorizados (16%).

Con relación al número de publicaciones recibidas y generadas por país entre 2019 y 2024. El primer valor se obtuvo dividiendo las publicaciones por país de origen o edición de la revista, procedimiento o libro. De acuerdo con esto, se ha establecido que los países con el mayor número de publicaciones recibidas con relación al alcance del SCA son Canadá, Estados Unidos, Reino Unido y Holanda, entre otros. Para registrar las publicaciones según el país de origen, se determinó la universidad, centro de investigación u organización a la que están afiliados los autores. El país de generación se determinó principalmente utilizando el criterio más común entre todos los autores del

trabajo y, en algunos casos, el lugar de origen del primer autor. Como resultado, hemos descubierto que Canadá, los Estados Unidos, China y Japón son los países con la mayor cantidad de publicaciones generadas en base a SCA. La tabla 1 se muestran los recursos donde se publicaron más investigaciones sobre enfermedades cardiovasculares, en particular las relacionadas con el síndrome coronario agudo (SCA).

| Revista                                              | Publicaciones | País           | ISSN      | SJR | JCR |
|------------------------------------------------------|---------------|----------------|-----------|-----|-----|
| <i>Canadian Journal of Cardiology</i>                | 65            | Canada         | 1916-7075 | Q1  | Q1  |
| <i>American Journal of Cardiology</i>                | 16            | United States  | 1879-1913 | Q1  | Q2  |
| <i>IEEE Access</i>                                   | 9             | United States  | 2169-3536 | Q1  | Q1  |
| <i>BMC BIOINFORMATICS</i>                            | 5             | United Kingdom | 1471-2105 | Q1  | NO  |
| <i>Artificial Intelligence in Medicine</i>           | 5             | Netherlands    | 1873-2860 | Q1  | Q1  |
| <i>Trends in Cardiovascular Medicine</i>             | 5             | United States  | 1050-1738 | Q1  | Q1  |
| <i>Journal of Biomedical Informatics</i>             | 4             | United States  | 1532-0464 | Q1  | Q2  |
| <i>Journal of the American College of Cardiology</i> | 4             | United States  | 0735-1097 | Q1  | Q1  |

Tabla 1 – Revistas con más publicaciones

## 5. Análisis técnico

### 5.1. Categorías de análisis técnico

Las variables que influyen directamente en la caracterización temprana de biomarcadores y las tendencias en el uso de herramientas tic para la identificación, evaluación y predicción del riesgo cardiovascular, se identificaron al revisar los artículos seleccionados y tener en cuenta los marcadores de investigación. En este sentido, se han identificado 245 artículos científicos para el desarrollo de la investigación, con relación a la RSL y cadenas de búsqueda aplicadas, por lo que se han agrupado en cinco parámetros técnicos para su análisis: Caracterización del síndrome coronario agudo (SCA), enfermedad cardiovascular distinta del síndrome agudo de la enfermedad de las arterias coronarias (OCD), uso de tecnología para la identificación de enfermedades cardiovasculares (AITC) y Sistema de Evaluación de Riesgo Cardiovascular (RTCS). Se encontraron 62 artículos científicos para la caracterización del síndrome coronario agudo con una proporción del 25% del total de artículos identificados, de los 46 artículos científicos encontrados para enfermedades cardiovasculares diferentes al síndrome coronario agudo, con una proporción del 19%, 95 artículos científicos para aplicación de tecnología en la identificación de enfermedades del corazón con una proporción del 39% del total de artículos identificados y para el sistema de evaluación de riesgo cardiovascular se encontrar 42 artículos con una proporción del 17% del total de artículos identificados.

### **5.1.1. Caracterización del síndrome coronario agudo**

Se encontraron 62 artículos para la caracterización del síndrome coronario agudo (SCA) y se dividieron en tres categorías: Identificación y Caracterización de patrones asociados al Síndrome Coronario Agudo (ICSCA), Clasificación del síndrome coronario agudo y patologías clínicas asociadas (CSCA), Historia clínica electrónica-EMR y Conjunto de datos clínicos tipo historia clínica electrónica (KDD). Se identificaron 27 artículos para ICSCA con una proporción del 44 %, 23 artículos para CSCA con una proporción del 37% y 12 para KDD con una proporción del 19%. En el análisis de doce comorbilidades de muestras de pacientes con tendencia médica al infarto de miocardio (Gulea, 2021; Kramer, 2019; Ordovas, 2021; Retnakaran, 2021), se examinó la viabilidad de utilizar métodos analíticos como los análisis de correlación PCA (Citro, 2020), LCA (Leonardi, 2021), Pearson (Harhash, 2021) e ICA (Khan, 2019) para clasificar la tendencia a sufrir infarto de miocardio solo basándose en las comorbilidades previas al evento cardíaco. En contraste, un estudio cuyo objetivo fue investigar en el ensayo MATRIX (Minimizing Adverse Hemorrhagic Events by Transradial Access Site and Systemic Implementation of Angiox) la incidencia, los predictores y las implicaciones pronósticas de la caída y los cambios de hemoglobina en pacientes con SCA manejados invasivamente estratificados por la presencia de sangrado intrahospitalario.

Además, en (Knowles, 2018) identifican los datos clínicos iniciales utilizados tradicionalmente para sospechar un SCA en pacientes con DTN en el servicio de urgencias; todos identifican a los pacientes con mayor riesgo de ser clasificados inicial y finalmente como SCA; algunos sobrestiman y otros subestiman inicialmente el riesgo final. De manera similar, en (Al Ghorani, 2021; Andreu, 2020; Lu, 2021; Palojoki, 2021) encuentran bioindicadores prometedores para enfermedades cardiovasculares agudas. Se ha demostrado en numerosos estudios clínicos que existe una correlación significativa entre los valores de sST2 y los resultados del paciente. Es importante destacar que se ha encontrado una correlación entre los niveles más altos de sST2 y un mayor riesgo de remodelado ventricular izquierdo adverso. Como resultado, sST2 podría ser una herramienta útil para la estratificación de riesgo y el diagnóstico de pacientes que ingresan al servicio de urgencias.

### **5.1.2. Identificación de enfermedades cardiovasculares distintas del síndrome coronario agudo (SCA)**

Los artículos de la investigación se dividieron en dos pilares o categorías: Descripción de enfermedades cardiovasculares distintas a la SCA (DOSCAD) y Criterios para la Identificación de Patologías (CIP). Se encontraron 24 artículos para Descripción de enfermedades cardiovasculares diferentes a la SCA (DOSCAD) con una relación del 52%, mientras que 22 artículos para criterios para la identificación de patologías (CIP) con una relación del 48%. En (Leslie, 2020) se realizó un estudio observacional prospectivo (ARTEMIS) en pacientes con síntomas lipídicos anormales para identificar y comparar la incidencia de muerte súbita cardíaca por diabetes y pacientes no diabéticos con infarto de miocardio. El mapeo electrofisiológico de alta densidad también se utilizó en (Brun, 2020; Hong, 2019) para detectar y caracterizar fenotípicamente la fibrilación ventricular idiopática en pacientes de edad avanzada.

### **5.1.3. Identificación de enfermedades cardiovasculares distintas del síndrome coronario agudo (SCA)**

Se encontraron 95 artículos que se dividieron en cuatro categorías para el desarrollo de aplicaciones tecnológicas enfocadas en la identificación de enfermedades cardiovasculares: implementación de técnicas de aprendizaje automático para la predicción de enfermedades cardiovasculares diferentes a la SCA a nivel de pre-evento cardíaco (MLTSCA), de técnicas de aprendizaje automático para la predicción de SCA a nivel de pre-evento cardíaco (MLTSCABE), Problemas y limitaciones en conjuntos de datos (PLKDD) y Propuestas basadas en modelos estadísticos (OM). Se identificaron 32 artículos para la implementación de técnicas de aprendizaje automático para la predicción de enfermedades cardiovasculares diferentes a la SCA a nivel de pre-evento cardíaco (MLTSCA), que representaban el 34% de los artículos encontrados, mientras que 26 artículos para la implementación de técnicas de aprendizaje automático para la predicción de SCA a nivel de pre-evento cardíaco (MLTSCABE) con una proporción del 27% de los artículos encontrados. Para problemas y limitaciones en conjuntos de datos (PLKDD) se encontró un total de 18 artículos con una proporción del 19%, y para las propuestas basadas en modelos estadísticos (OM), un total de 19 artículos científicos con una proporción de 20%.

Algunas investigaciones relacionadas con la identificación de patrones (Blagova, 2019; Haïssaguerre, 2020), el preprocesamiento de datos (Dinesh, 2018), las técnicas de eliminación de datos ruidosos (Al'Aref, 2019), la eliminación de datos faltantes y la clasificación de atributos para la predicción (Mezzatesta, 2019) y la toma de decisiones como herramientas basadas en tecnologías de la información enfocadas en inteligencia artificial (Dinh, 2019; Karunathilake; 2018; Seetharam, 2019) y la construcción de modelos para el diagnóstico oportuno de enfermedades asociadas al síndrome coronario agudo son el método más relevante y utilizado para la identificación temprana de cardiopatías con especial atención a las enfermedades asociadas al síndrome coronario agudo (Ramalingam, 2018).

### **5.1.4. Sistema de evaluación del riesgo cardiovascular**

Se encontraron 42 artículos para encontrar sistemas de evaluación del riesgo cardiovascular, de los cuales 3 pertenecían a tres categorías: Modelos enfocados a la identificación del riesgo cardiovascular (mundial) (MCVG), Modelos enfocados a la identificación del riesgo cardiovascular (Colombia) (MCVC) y Análisis de datos para la identificación (DAI). Se identificaron 21 artículos para los modelos enfocados a la identificación del riesgo cardiovascular (mundial) (MCVG), que representaron el 50% del total categorizado; 10 artículos para los modelos enfocados a la identificación del riesgo cardiovascular en Colombia (MCVC), que representaron el 24% del total encontrado; y 11 artículos para el Análisis de datos para la identificación (DAI), que representaron el 26% del total de artículos.

Algunas investigaciones relacionadas con los sistemas de evaluación del riesgo cardiovascular se centran en el análisis estadístico de la identificación de enfermedades utilizando diferentes métodos como SCORE (Everett, 2019), TIMI (Kao, 2020) y FRAMINGHAM (Komiyama, 2018) como métodos más efectivos para la identificación de dichas cardiopatías a nivel pre-evento cardíaco. De manera similar, en (Kao, 2020)

se explican diferentes herramientas para medir el riesgo cardiovascular donde existen marcadores de riesgo que aportan poco a la estratificación y las más utilizadas son las tablas de estratificación de riesgo. Además, es importante mencionar que las tablas de riesgo pueden evaluar y controlar preventivamente la hipertensión arterial y la dislipidemia (Chotechuang, 2020; Gernaat, 2018; Komiyama, 2018).

## **5.2. Conjuntos de datos y técnicas de minería de datos aplicadas para la predicción de enfermedades cardiovasculares**

La minería de datos es el proceso de encontrar patrones y tendencias previamente desconocidos en las bases de datos y usar esa información para construir modelos predictivos (Murphy, 2020). En salud, la minería de datos es un campo de gran importancia y se ha vuelto cada vez más eficiente e imprescindible. En la actualidad, la industria de la salud genera grandes cantidades de datos complejos sobre pacientes, recursos hospitalarios, diagnósticos de enfermedades, registros electrónicos de pacientes, dispositivos, etc. La gran cantidad de datos es un recurso clave para ser procesado y analizado para la extracción de conocimiento y permitir el ahorro de costos, y toma de decisiones. La minería de datos proporciona un conjunto de herramientas y técnicas que se pueden aplicar a los datos para lograr estos objetivos (Lloyd-Jones, 2019).

Los conjuntos de datos que tratan los mismos problemas médicos, como la enfermedad de las arterias coronarias (CAD) (Alić, 2017), pueden mostrar resultados diferentes cuando se aplica la misma técnica de aprendizaje automático. Los resultados de precisión de la clasificación y las características importantes seleccionadas se basan principalmente en la eficiencia del diagnóstico y análisis médico. Estos conjuntos de datos brindan información complementaria al diagnóstico médico y, por lo tanto, pueden mejorar las tasas de precisión de la clasificación (Fatima, 2017; Pouriyeh, 2017; Nilashi, 2017; Ramalingam, 2018). El diagnóstico de los pacientes se basa en la precisión de los datos médicos recopilados. Las características comunes entre estos conjuntos de datos se extraen y utilizan en análisis posteriores para la misma enfermedad en cualquier conjunto de datos.

En (Abdar, 2019; Dwivedi, 2018), se denotan los conjuntos de datos con mayor implementación en áreas de construcción de modelos de inteligencia artificial, los cuales son elegidos por su efectividad, tales como: Cleveland Clinic Foundation, Instituto Húngaro de Cardiología, V.A. Medical Center, Long Beach, CA y registro de estadísticas del proyecto. El número de instancias para cada conjunto de datos es el siguiente: Cleveland: 303, Húngary: 294, Long Beach VA: 200 y Project Statlog: 270. Estos conjuntos de datos contienen 14 atributos que se extrajeron de un conjunto más grande de 75 atributos. Estos conjuntos de datos públicos se pueden descargar del repositorio de UCI para su implementación (The UCI Machine Learning Repository, n. d.).

## **6. Conclusión y trabajos futuros**

### **6.1. Conclusión**

El propósito de este artículo ha sido proponer una serie de recomendaciones para la investigación centrada en las enfermedades cardiovasculares en relación con la

identificación del conjunto de datos más adecuado según el tipo de investigación. El desarrollo en esta área de investigación en los últimos 5 años ha crecido mucho desde la presentación de 19 artículos de investigación en 2019 a 69 artículos publicados en 2023. La base de datos especializada más representativa, con relación al número de publicaciones científicas en SCA es Science Direct con un 68,16% de las publicaciones en total (167) superando la suma de los resultados obtenidos al consultar las bases de datos Wos, PubMed y BioMed.

Es importante resaltar que la mayor parte de las publicaciones en este campo de investigación son revistas con 208 artículos científicos con una relación total del 84,89% de las publicaciones encontradas. donde un porcentaje muy importante se realiza en revistas del primer (55% con 135 artículos) y segundo cuartil (20% con 49 artículos). Canadá y Estados Unidos se destacan como los países con mayor número de publicaciones aceptadas en esta área de investigación, con una participación de 26.94% para Canadá y 24.49% para Estados Unidos con relación al total de publicaciones recibidas de nivel mundial.

Aunque los medios en los que se publican los resultados de la investigación en esta área del conocimiento son muy diversos, los más destacados son: Canadian Journal of Cardiology (Canadá) con un 26,54% y American Journal of Cardiology (EE.UU.) con un 6,53%. También, se logró identificar que las instituciones con mayor experiencia en investigación en este campo son: Montreal Behavioral Medicine Centre, CIUSSS-NIM (Canadá) con una participación del 12,25% y Department of Health, Kinesiology, and Applied Physiology (HKAP), Concordia University (CANADÁ) con una participación del 8,57% respectivamente.

## 6.2. Trabajos futuros

Las tecnologías de la información y su contribución a la identificación de enfermedades cardiovasculares mediante la implementación de técnicas de inteligencia artificial es una de las soluciones que ha despegado en estos tiempos. En este sentido, se postulan los siguientes trabajos futuros: conjunto de datos clínicos basados en muestras médicas no invasivas, construcción de modelos de inteligencia artificial basados en redes neuronales artificiales, desarrollo de un sistema basado en el método wrapper donde se hibridan técnicas de selección de características. como chi cuadrado, ganancia de información, relación de ganancia con técnicas de soporte de máquina vectorial para crear un sistema con un alto nivel de precisión enfocado a la predicción de enfermedades cardiovasculares

## Referencias

- Abdar, M., Książek, W., Acharya, U. R., Tan, R. S., Makarenkov, V., & Plawiak, P. (2019). A new machine learning technique for an accurate diagnosis of coronary artery disease. *Computer Methods and Programs in Biomedicine*, 179, 104992.
- Al'Aref, S. J., Anchouche, K., Singh, G., Slomka, P. J., Kolli, K. K., Kumar, A., ... & Min, J. K. (2019). Clinical applications of machine learning in cardiovascular disease and its relevance to cardiac imaging. *European Heart Journal*, 40(24), 1975-1986.

- Al Ghorani, H., Schwarz, V., Vukadinovic, D., Fries, P., Scheller, B., & Mahfoud, F. (2021). Myocardial infarction in a patient with single coronary artery-rare but real. *Journal of Cardiology Cases*, 23(5), 246-249.
- Aleksova, A., Paldino, A., Beltrami, A. P., Padoan, L., Iacoviello, M., Sinagra, G., ... & Maisel, A. S. (2019). Cardiac biomarkers in the emergency department: the role of soluble ST2 (sST2) in acute heart failure and acute coronary syndrome—there is meat on the bone. *Journal of Clinical Medicine*, 8(2), 270.
- Alić, B., Gurbeta, L., & Badnjević, A. (2017). Machine learning techniques for classification of diabetes and cardiovascular diseases. In 2017 6th Mediterranean Conference on Embedded Computing (MECO) (pp. 1-4). IEEE.
- Andreu, Ó. M., Nadal, G. M., Hernández, S. J., Gómez-Angelats, E., Alonso, J. R., Santaliestra, A. A., ... & Barbeito, B. L. (2020). Asociación entre los datos clínicos y electrocardiográficos iniciales en pacientes con dolor torácico no traumático y la sospecha inicial y el diagnóstico final de síndrome coronario agudo. *Emergencias: Revista de la Sociedad Española de Medicina de Urgencias y Emergencias*, 32(1), 9-18.
- Blagova, O., Osipova, Y., Nedostup, A., Kogan, E., Zaitsev, A., & Fomin, V. (2019). Diagnostic value of different noninvasive criteria of latent myocarditis in comparison with myocardial biopsy. *Cardiology*, 142(3), 167-174.
- Brandberg, H., Kahan, T., Spaak, J., Sundberg, K., Koch, S., Adeli, A., ... & Zakim, D. (2020). A prospective cohort study of self-reported computerised medical history taking for acute chest pain: protocol of the CLEOS-Chest Pain Danderyd Study (CLEOS-CPDS). *BMJ Open*, 10(1).
- Brun, F., Gigli, M., Graw, S. L., Judge, D. P., Merlo, M., Murray, B., ... & James, C. A. (2020). FLNC truncations cause arrhythmogenic right ventricular cardiomyopathy. *Journal of Medical Genetics*, 57(4), 254-257.
- Chotechuang, Y., Phrommintikul, A., Kuanprasert, S., Muenpa, R., Ruengorn, C., Patumanond, J., ... & Sukonthasarn, A. (2020). GRACE score and cardiovascular outcomes prediction among the delayed coronary intervention after post-fibrinolytic STEMI patients in a limited PCI-capable hospital. *Open Heart*, 7(1), e001133.
- Chu, D., Al Rifai, M., Virani, S. S., Brawner, C. A., Nasir, K., & Al-Mallah, M. H. (2020). The relationship between cardiorespiratory fitness, cardiovascular risk factors, and atherosclerosis. *Atherosclerosis*.
- Citro, R., Okura, H., Ghadri, J. R., Izumi, C., Meimoun, P., Izumo, M., ... & EACVI Scientific Documents Committee. (2020). Multimodality imaging in takotsubo syndrome: a joint consensus document of the European Association of Cardiovascular Imaging (EACVI) and the Japanese Society of Echocardiography (JSE). *European Heart Journal-Cardiovascular Imaging*, 21(11), 1184-1207. <https://doi.org/10.1093/ehjci>

- Comas-Gonzalez, Z., Mardini-Bovea, J., Salcedo, D., & De-la-Hoz-Franco, E. (2023). Comparison of Supervised Techniques of Artificial Intelligence in the Prediction of Cardiovascular Diseases. In H. Degen, S. Ntoa, & A. Moallem (Eds.), *HCI International 2023 – Late Breaking Papers. HCII 2023. Lecture Notes in Computer Science*, vol 14059. Springer, Cham. [https://doi.org/10.1007/978-3-031-48057-7\\_4](https://doi.org/10.1007/978-3-031-48057-7_4)
- Crimi, G., Mandurino-Mirizzi, A., Gritti, V., Scotti, V., Strozzi, C., De Silvestri, A., ... & Klersy, C. (2020). Percutaneous coronary intervention techniques for bifurcation disease: network meta-analysis reveals superiority of double-kissing crush. *Canadian Journal of Cardiology*, 36(6), 906-914.
- Dinesh, K. G., Arumugaraj, K., Santhosh, K. D., & Mareeswari, V. (2018). Prediction of cardiovascular disease using machine learning algorithms. In 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT) (pp. 1-7). IEEE.
- Dinh, A., Miertschin, S., Young, A., & Mohanty, S. D. (2019). A data-driven approach to predicting diabetes and cardiovascular disease with machine learning. *BMC Medical Informatics and Decision Making*, 19(1), 1-15.
- Dwivedi, A. K. (2018). Performance evaluation of different machine learning techniques for prediction of heart disease. *Neural Computing and Applications*, 29(10), 685-693.
- Everett, C. C., Fox, K. A., Reynolds, C., Fernandez, C., Sharples, L., Stocken, D. D., ... & Gale, C. P. (2019). Evaluation of the impact of the GRACE risk score on the management and outcome of patients hospitalised with non-ST elevation acute coronary syndrome in the UK: protocol of the UKGRIS cluster-randomised registry-based trial. *BMJ Open*, 9(9), e032165.
- Fatima, M., & Pasha, M. (2017). Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, 9(01), 1.
- García-Pérez, L. (2013). Systematic review of health-related utilities in Spain: the case of mental health. *Gaceta Sanitaria*, 28(1), 77-83. <https://doi.org/10.1016/j.gaceta.2013.04.006>
- Gernaat, S. A., Boer, J. M., van den Bongard, D. H., Maas, A. H., van der Pol, C. C., Bijlsma, R. M., ... & Peeters, P. H. (2018). The risk of cardiovascular disease following breast cancer by Framingham risk score. *Breast Cancer Research and Treatment*, 170(1), 119-127.
- González-Alcaide, G., Llorente, P., & Ramos-Rincón, J. M. (2020). Systematic analysis of the scientific literature on population surveillance. *Heliyon*, 6(10), e05141.
- Grams, M. E. (2011). Validation of CKD and related conditions in existing data sets: A systematic review. *American Journal of Kidney Diseases*, 57(1), 44-54. <https://doi.org/10.1053/j.ajkd.2010.05.013>

- Gulea, C., Zakeri, R., & Quint, J. K. (2021). Model-based comorbidity clusters in patients with heart failure: association with clinical outcomes and healthcare utilization. *BMC Medicine*, 19, 9. <https://doi.org/10.1186/s12916-020-01881-7>
- Haïssaguerre, M., Duchateau, J., Dubois, R., Hocini, M., Cheniti, G., Sacher, F., ... & Bernus, O. (2020). Idiopathic ventricular fibrillation: role of Purkinje system and microstructural myocardial abnormalities. *Clinical Electrophysiology*, 6(6), 591-608.
- Harhash, A. A., May, T. L., Hsu, C. H., Agarwal, S., Seder, D. B., Mooney, M. R., ... & Kern, K. B. (2021). Risk stratification among survivors of cardiac arrest considered for coronary angiography. *Journal of the American College of Cardiology*, 77(4), 360-371.
- Helwani, M. A., Amin, A., Lavigne, P., Rao, S., Oesterreich, S., Samaha, E., ... & Nagele, P. (2018). Etiology of acute coronary syndrome after noncardiac surgery. *Anesthesiology*, 128(6), 1084-1091.
- Henriquez, C. (2022). Predictive Model of Cardiovascular Diseases Implementing Artificial Neural Networks. In K. Saeed & J. Dvorský (Eds.), *Computer Information Systems and Industrial Management. CISIM 2022. Lecture Notes in Computer Science*, vol 13293. Springer, Cham. [https://doi.org/10.1007/978-3-031-10539-5\\_17](https://doi.org/10.1007/978-3-031-10539-5_17)
- Hong, N., Wen, A., Stone, D. J., Tsuji, S., Kingsbury, P. R., Rasmussen, L. V., ... & Jiang, G. (2019). Developing a FHIR-based EHR phenotyping framework: A case study for identification of patients with obesity and multiple comorbidities from discharge summaries. *Journal of Biomedical Informatics*, 99, 103310.
- Iannattone, P. A., Zhao, X., VanHouten, J., Garg, A., & Huynh, T. (2020). Artificial intelligence for diagnosis of acute coronary syndromes: a meta-analysis of machine learning approaches. *Canadian Journal of Cardiology*, 36(4), 577-583.
- Idris, N. M., Chiam, Y. K., Varathan, K. D., Ahmad, W. A. W., Chee, K. H., & Liew, Y. M. (2020). Feature selection and risk prediction for patients with coronary artery disease using data mining. *Medical & Biological Engineering & Computing*, 58(12), 3123-3140. <https://doi.org/10.1007/s11517-020-02228-9>
- Instituto Nacional de Salud. (2023). 29 de septiembre - Día Mundial del Corazón. Recuperado el 5 de abril de 2024, de <https://www.ins.gov.co/Noticias/Paginas/29-de-septiembre---Dia-Mundial-del-Corazon.aspx>
- Kandasamy, S., & Anand, S. S. (2018). Cardiovascular disease among women from vulnerable populations: a review. *Canadian Journal of Cardiology*, 34(4), 450-457.
- Kao, Y. T., Hsieh, Y. C., Hsu, C. Y., Huang, C. Y., Hsieh, M. H., Lin, Y. K., & Yeh, J. S. (2020). Comparison of the TIMI, GRACE, PAMI and CADILLAC risk scores for prediction of long-term cardiovascular outcomes in Taiwanese diabetic patients with ST-segment elevation myocardial infarction: From the registry of the Taiwan Society of Cardiology. *PLoS ONE*, 15(2), e0229186.

- Karunathilake, S. P., & Ganegoda, G. U. (2018). Secondary prevention of cardiovascular diseases and application of technology for early diagnosis. *BioMed Research International*, 2018.
- Khan, M. U., Aziz, S., Iqtidar, K., Zainab, A., & Saud, A. (2019). Prediction of Acute Coronary Syndrome Using Pulse Plethysmograph. In 2019 4th International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEEST) (pp. 1-6). IEEE.
- Knowles, J. W., & Ashley, E. A. (2018). Cardiovascular disease: The rise of the genetic risk score. *PLoS Medicine*, 15(3), e1002546. <https://doi.org/10.1371/journal.pmed.1002546>
- Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering — A systematic literature review. *Information and Software Technology*, 51(1), 7–15. <https://doi.org/10.1016/j.infsof.2008.09.009>
- Komiyama, K., Nakamura, M., Tanabe, K., Niikura, H., Fujimoto, H., Oikawa, K., ... & Tokyo CCU Network Scientific Committee. (2018). In-hospital mortality analysis of Japanese patients with acute coronary syndrome using the Tokyo CCU Network database: Applicability of the GRACE risk score. *Journal of Cardiology*, 71(3), 251-258.
- Konys, A. (2019). Knowledge repository of ontology learning tools from text. *Procedia Computer Science*, 159, 1614-1628.
- Kramer, A. I., Trinder, M., & Brunham, L. R. (2019). Estimating the prevalence of familial hypercholesterolemia in acute coronary syndrome: a systematic review and meta-analysis. *Canadian Journal of Cardiology*, 35(10), 1322-1331.
- Kutyrev, K., Yakovlev, A., & Metsker, O. (2019). Mortality Prediction Based on Echocardiographic Data and Machine Learning: CHF, CHD, Aneurism, ACS Cases. *Procedia Computer Science*, 156, 97-104.
- Leonardi, S., Gragnano, F., Carrara, G., Gargiulo, G., Frigoli, E., Vranckx, P., ... & Valgimigli, M. (2021). Prognostic implications of declining hemoglobin content in patients hospitalized with acute coronary syndromes. *Journal of the American College of Cardiology*, 77(4), 375-388.
- Leslie, H. (2020). openEHR archetype use and reuse within multilingual clinical data sets: case study. *Journal of Medical Internet Research*, 22(11), e23361.
- Lloyd-Jones, D. M., Braun, L. T., Ndumele, C. E., Smith Jr, S. C., Sperling, L. S., Virani, S. S., & Blumenthal, R. S. (2019). Use of risk assessment tools to guide decision-making in the primary prevention of atherosclerotic cardiovascular disease: a special report from the American Heart Association and American College of Cardiology. *Circulation*, 139(25), e1162-e1177.

- Lu, J., Wang, L., Bennamoun, M., Ward, I., An, S., Sohel, F., ... & Sanfilippo, F. M. (2021). Machine learning risk prediction model for acute coronary syndrome and death from use of non-steroidal anti-inflammatory drugs in administrative data. *Scientific Reports*, 11(1), 1-10.
- Magoev, K., Krzhizhanovskaya, V. V., & Kovalchuk, S. V. (2018). Application of clustering methods for detecting critical acute coronary syndrome patients. *Procedia Computer Science*, 136, 370-379.
- Manterola, C., Astudillo, P., Arias, E., & Claros, N. (2013). Systematic reviews of the literature: What should be known about them. *Cirugía Española*, 91(3), 149-155. <https://doi.org/10.1016/j.ciresp.2011.07.009>
- Merlano-Porras, C. A., & Gorbanev, L. (2013). Health system in Colombia: A systematic review of literature. *Revista Gerencia Políticas Salud*, 12(24), 74-86.
- Mendes, D., Paredes, S., Rocha, T., Carvalho, P., Henriques, J., & Morais, J. (2018). Integration of current clinical knowledge with a data driven approach: An innovative perspective. *International Journal of Information Technology & Decision Making*, 17(01), 133-153.
- Metsker, O., Efimov, E., Trofimov, E., Kopanitsa, G., Bolgova, E., & Yakovlev, A. (2020). Performance improvement algorithms in big data analysis. *Procedia Computer Science*, 178, 386-393.
- Metsker, O., Bolgova, E., Yakovlev, A., Funkner, A., & Kovalchuk, S. (2017). Pattern-based mining in electronic health records for complex clinical process analysis. *Procedia Computer Science*, 119, 197-206.
- Mezzatesta, S., Torino, C., De Meo, P., Fiumara, G., & Vilasi, A. (2019). A machine learning-based approach for predicting the outbreak of cardiovascular diseases in patients
- Murphy, S. P., McCarthy, C. P., Cohen, J. A., Rehman, S., Jones-O'Connor, M., Olshan, D. S., ... & Wasfy, J. H. (2020). Application of the GRACE, TIMI, and TARRACO risk scores in type 2 myocardial infarction. *Journal of the American College of Cardiology*, 75(3), 344-345.
- Narayan, S. M., Wang, P. J., & Daubert, J. P. (2019). New concepts in sudden cardiac arrest to address an intractable epidemic: JACC state-of-the-art review. *Journal of the American College of Cardiology*, 73(1), 70-88. <https://doi.org/10.1016/j.jacc.2018.09.068>
- Nilashi, M., bin Ibrahim, O., Ahmadi, H., & Shahmoradi, L. (2017). An analytical method for diseases prediction using machine learning techniques. *Computers & Chemical Engineering*, 106, 212-223.
- Ordovas, K. G., Baldassarre, L. A., Bucciarelli-Ducci, C. (2021). Cardiovascular magnetic resonance in women with cardiovascular disease: position statement from the Society for Cardiovascular Magnetic Resonance (SCMR). *Journal of Cardiovascular Magnetic Resonance*, 23, 52. <https://doi.org/10.1186/s12968-021-00746-z>

- Organización Mundial de la Salud. (2023). Enfermedades cardiovasculares (ECV). Recuperado el 5 de abril de 2024, de [https://www.who.int/es/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/es/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds))
- Organización Panamericana de la Salud. (2023). Hipertensión. Recuperado el 5 de abril de 2024, de <https://www.paho.org/es/temas/hipertension>
- Palojoki, S., Saranto, K., Reponen, E., Skants, N., Vakkuri, A., & Vuokko, R. (2021). Classification of Electronic Health Record–Related Patient Safety Incidents: Development and Validation Study. *JMIR Medical Informatics*, 9(8), e30470.
- Pouriyeh, S., Vahid, S., Sannino, G., De Pietro, G., Arabnia, H., & Gutierrez, J. (2017). A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease. In 2017 IEEE Symposium on Computers and Communications (ISCC) (pp. 204-207). IEEE.
- Ramalingam, V. V., Dandapath, A., & Raja, M. K. (2018). Heart disease prediction using machine learning techniques: a survey. *International Journal of Engineering & Technology*, 7(2), 684-687.
- Retnakaran, R. (2018). Novel biomarkers for predicting cardiovascular disease in patients with diabetes. *Canadian Journal of Cardiology*, 34(5), 624-631.
- Rezaianzadeh, A., Dastoorpoor, M., Sanaei, M., Salehnasab, C., Mohammadi, M. J., & Mousavizadeh, A. (2020). Predictors of length of stay in the coronary care unit in patient with acute coronary syndrome based on data mining methods. *Clinical Epidemiology and Global Health*, 8(2), 383-388.
- Sanchez, A., Neira, D., & Cabello, J. J. (2016). Frameworks applied in quality management—A systematic review. *Rev. Espacios*, 37(9), 17.
- Seetharam, K., Shrestha, S., & Sengupta, P. P. (2019). Artificial intelligence in cardiovascular medicine. *Current Treatment Options in Cardiovascular Medicine*, 21(5), 1-14.
- Si, Y., Bernstam, E. V., & Roberts, K. (2021). Generalized and transferable patient language representation for phenotyping with limited data. *Journal of Biomedical Informatics*, 116, 103726.
- Sun, Q., Wang, Q., Ji, B., Wu, W., Huang, W., & Wang, C. (2020). The Cardiodynamicsgram Based Early Detection of Myocardial Ischemia Using the Lempel-Ziv Complexity. *IEEE Access*, 8, 207894-207904.
- The UCI Machine Learning Repository. (n. d.). Heart Disease Data Set. Retrieved from <https://archive.ics.uci.edu/ml/datasets/Heart+Disease>
- Yao, L., Liu, C., Li, P., Wang, J., Liu, Y., Li, W., ... & Zhang, H. (2020). Enhanced automated diagnosis of coronary artery disease using features extracted from QT interval time series and ST-T waveform. *IEEE Access*, 8, 129510-129524. <https://doi.org/10.1109/ACCESS.2020.3001787>

- Xie, L., Li, Z., Zhou, Y., He, Y., & Zhu, J. (2020). Computational Diagnostic Techniques for Electrocardiogram Signal Analysis. *Sensors*, 20(21), 6318.
- Xue, Y., Hu, Z., Jing, Y., Wu, H., Li, X., Wang, J., ... & Lv, Q. (2020). Efficacy assessment of ticagrelor versus clopidogrel in Chinese patients with acute coronary syndrome undergoing percutaneous coronary intervention by data mining and machine-learning decision tree approaches. *Journal of Clinical Pharmacy and Therapeutics*, 45(5), 1076-1086.

# Internet de las cosas aplicado a la agricultura: estado actual y su aplicación mediante un prototipo

Jorge Luis Diaz Martinez<sup>1</sup>, Dixon Salcedo<sup>1</sup>, Teobaldis Mercado<sup>2</sup>, Yadira Quiñonez<sup>3</sup>,  
Andrés Mejía de la Hoz<sup>1</sup>

[jdiaz5@cuc.edu.co](mailto:jdiaz5@cuc.edu.co); [dsalcedo2@cuc.edu.co](mailto:dsalcedo2@cuc.edu.co); [temercado@unicordoba.edu.co](mailto:temercado@unicordoba.edu.co);  
[yadiraqui@uas.edu.mx](mailto:yadiraqui@uas.edu.mx); [emejia17@cuc.edu.co](mailto:emejia17@cuc.edu.co)

<sup>1</sup> Universidad de la Costa-CUC, Departamento de Ciencias de la Computación y Electrónica, 08001, Atlántico, Colombia.

<sup>2</sup> Universidad de Córdoba. Departamento de ingeniería Agronómica, 230002, Montería, Córdoba, Colombia.

<sup>3</sup> Universidad Autónoma de Sinaloa. Facultad de Informática de Mazatlán, 82000, Mazatlán, Sinaloa, México.

DOI: 10.17013/risti.53.106–121

**Resumen:** El sector agrícola se encuentra entre los más beneficiados por la expansión del Internet de las Cosas (IoT), ya que permite recopilar y gestionar una gran cantidad de datos sobre el entorno de un cultivo. Este artículo presenta los resultados del diseño y la prueba de un prototipo de sistema de monitoreo de variables agrícolas, las cuales pueden almacenarse y consultarse en la nube. Se utilizaron sensores y sistemas de procesamiento de bajo costo y fácilmente adaptables a entornos agrícolas. Los resultados obtenidos confirman la viabilidad de implementar estos sistemas sin recurrir a herramientas tecnológicas y computacionales complejas.

**Palabras-clave:** Agricultura de precisión, dashboard, ESP32, IoT, sensores, monitoreo en la nube.

## *Internet of Things (IoT) applied to agriculture: current state and its application through a prototype*

**Abstract:** The agricultural sector is among the most benefited by the Internet of Things (IoT) expansion since it allows collecting and managing a large amount of data about a crop's environment. This article presents the results of designing and testing a prototype monitoring system for agricultural variables, which can be stored and consulted in the cloud. Low-cost sensors and processing systems were easily adaptable to agricultural environments. The results confirmed the viability of implementing this system without resorting to complex technological and computational tools.

**Keywords:** Precision agriculture, dashboard, ESP32, IoT, sensors, cloud monitoring.

## 1. Introducción

Resulta evidente la importancia de la agricultura en la vida humana, desde su aparición hace aproximadamente 12500 años. Este hecho impactó la significativamente el desarrollo del homo sapiens. Fue un proceso que supuso la domesticación de plantas y animales, lo cual permitió el asentamiento y conformación de grupos humanos (Bula, 2020). Por otra parte, la agricultura ha sido un instrumento para la erradicación de la pobreza y el hambre en muchas zonas del mundo, pues la agricultura es la responsable principal de la producción de alimentos.

Las tecnologías de la información y la comunicación (TIC) han impactado todos los aspectos de la vida humana. Desde los empleos hasta el entretenimiento pasando por la educación y otras necesidades tales como servicios bancarios, servicios de salud, entre otros (Arsovski et al., 2016; Bankole et al., 2011; Gašparović, 2019; Hernandez, 2017). Como era de esperar, la agricultura no ha sido ajena a dicho proceso de tecnificación, y la vinculación de las TIC ha dado lugar a lo que en este momento se denomina agricultura de precisión. La inclusión de tecnología a una de las actividades humanas más antiguas ha traído consigo mejoras en la productividad y garantías en la seguridad alimentaria de la población mundial (Ali, 2012; S. Das et al., 2017; Gopinath, n.d.; Marwa et al., 2020; Newase et al., 2017; Punchihewa & Wimalaratne, 2010; Thangaraju et al., 2016).

A pesar de las oportunidades que ofrece la tecnificación del sector agrícola, Colombia aún tiene un bajo nivel (Minagricultura, 2018; Parra-Peña et al., 2021). Debido a esta baja tecnificación del sector agrario colombiano, se carece de histórico de datos (*dataset*) para la región en lo que tiene que ver con la medición de variables agrícolas y climáticas, para poder realizar análisis fenológicos de los diferentes tipos de cultivos. Las variables climáticas no son importantes sólo por la fenología, sino por el impacto que tiene el clima en la correcta realización de las actividades del campo. Como un ejemplo de esto se puede mencionar los sistemas de alertas tempranas (SAT), que anuncian la ocurrencia de inundaciones, en especial en regiones de Colombia como el sur del Atlántico, que en la actualidad es inexistente, a pesar del conocido riesgo (UNGRD, 2021). La región correspondiente al sur del Atlántico toma relevancia por ser un importante productor de maíz.

El SAT se puede constituir en el primer paso para conseguir la tecnificación del sector rural colombiano, en especial, en la región Caribe. Se convertirían en el eje central en torno a cuál desplegar los otros sistemas complementarios que demanda la agricultura de precisión, relacionados con el monitoreo de cultivo, y la actuación en actividades productivas.

Cuando se revisa la literatura de forma global se pueden identificar 4 categorías de análisis de las TIC aplicadas a la agricultura, dando lugar a la agricultura de precisión. Según las tendencias observadas, a saber: IoT en la agricultura de precisión (Mohammad El-Basioni & Abd El-Kader, 2020); arquitecturas y servicios (Dobrescu et al., 2019); sistemas expertos en la toma de decisiones (Bouachir et al., 2019); monitoreo y salud de los cultivos (L. Liu et al., 2019). De estas categorías analizadas se desprenden oportunidades de investigación en diferentes ámbitos.

El primer ámbito está relacionado con los dataset aplicados en la agricultura de precisión. Debido a la heterogeneidad de los dataset, se hace necesaria la aplicación de técnicas predictivas basadas en analítica de datos (Big Data). Las técnicas de Big Data resultan las más adecuadas para el tratamiento de datos masivos y poco estructurados (Oussous et al., 2018). Por otro lado, la seguridad de los datos que son detectados por medio de dispositivos IoT debe establecerse mediante técnicas nuevas de protección, pues se trata de arquitectura no tradicionales (Amanullah et al., 2020; Khan & Salah, 2018; Mohamad Noor & Hassan, 2019; Sidhu et al., 2019). Finalmente, usar la información colectada en los dataset para identificar enfermedades y anomalías en los cultivos a fin de ampliar aún más el alcance de la agricultura de precisión (Bhatia et al., 2021).

El segundo ámbito está relacionado con la evaluación de la eficacia de los modelos predictivos. Son necesarios análisis comparativos que permitan evaluar y contrastar la capacidad predictiva de los modelos frente a la realidad, con el fin de validar su precisión y confiabilidad (de la Casa et al., 2018; Ge et al., 2020; Miller et al., 2019; Panesar et al., 2021; Vergara-Díaz et al., 2020; Wei et al., 2021).

El tercer ámbito que ofrece oportunidades es el de la inteligencia artificial. Los chatbots basados en inteligencia artificial al servicio de los agricultores, con la aplicación de técnicas de aprendizaje automático (ML, Machine Learning), aprendizaje profundo (DL, Deep Learning) y algoritmos híbridos. Los algoritmos de inteligencia artificial también ofrecen un enfoque más preciso y rentable de la agricultura, haciéndola más productiva y sostenible, especialmente en la actual situación de escasez de mano de obra y cambio climático (Alfred et al., 2021; García et al., 2020; Kocian & Incrocci, 2020; Z. Unal, 2020).

Esta investigación se realiza en el marco de un proyecto enfocado en los sistemas SAT aplicados al sector rural, para servir de apoyo en la ejecución de acciones de prevención y mitigación de riesgos de cambio climático, a los agricultores del sur del Atlántico, especialmente a los productores de maíz. Sin embargo, el análisis de la literatura que se realiza a continuación para identificar las necesidades tecnológicas del sector agrícola y las tecnologías de implementación de soluciones es global. Se debe tener en cuenta que gran parte de las variables que se miden para alimentar los SAT, también son utilizadas en las implementaciones de agricultura de precisión para operar los sistemas de monitoreo y control. Sin embargo, se le dedica una sección aparte a los SAT.

## 2. Metodología

En este artículo se ha realizado una revisión documental de la literatura relacionada con las necesidades de gestión que se encuentran en el campo de la agricultura de precisión, y las tecnologías a implementar para la satisfacción de esas necesidades. Para tal fin, se debe realizar una búsqueda sistemática de la literatura en las bases de datos siguiendo una metodología definida.

La metodología se desarrolla en cuatro pasos, siendo el primero de ellos la definición de los parámetros de búsqueda. Para la definición de tales parámetros se plantea inicialmente una pregunta de investigación: ¿Cuáles son las necesidades y las implementaciones tecnológicas que demanda el sector de la agricultura de precisión?

Como se necesitaba explorar todas las tecnologías aplicables al sector de la agricultura de precisión, las palabras clave que se extrajeron de la pregunta fueron 3: “Necesidades”, “Tecnologías” y “Agricultura de Precisión”. Las cadenas de búsqueda obtenidas fueron: “Agricultura de Precisión”, “Necesidades” AND “Agricultura de Precisión” y “Tecnologías” AND “Agricultura de Precisión”. Para las búsquedas en las bases de datos, se usaron las versiones en inglés de las palabras clave.

El segundo paso de la metodología es definir las bases de datos a consultar. El criterio utilizado es que las bases de datos ofrecieran acceso a texto completo de los artículos. De los recursos provistos por la Universidad de la Costa, las 2 que cumplen este criterio son IEEEExplore y ScienceDirect. Se descartaron Springer, Scopus y WoS al no proporcionar acceso a texto completo la mayoría de las veces. Se consultó también el buscador Google, para considerar algunos documentos que podrían llegar a ser importantes y no figuran en las bases de datos antes mencionadas. Ejemplos de tales documentos pueden ser los de tipo institucional, publicados por autoridades, ministerios, gobiernos, entre otros.

El tercer paso es realizar las búsquedas en las bases de datos seleccionadas. Hacer la identificación de los documentos, la verificación de acceso y la depuración de los artículos según las necesidades de esta investigación.

Finalmente, el cuarto paso consiste en realizar el análisis de los resultados obtenidos.

### 3. Análisis de resultados

Para rastrear la evolución de la Agricultura de Precisión (AP) y su creciente impacto, se efectuó una búsqueda exhaustiva del término “Precision Farming” en la base de datos IEEEExplore. Como se aprecia en la Figura 1, el interés en este tema en IEEE ha experimentado un crecimiento gradual, consolidándose a partir del año 2013. Esta tendencia pone de manifiesto el carácter emergente de la AP, lo que la convierte en un campo de investigación y desarrollo con un sinnúmero de posibilidades.

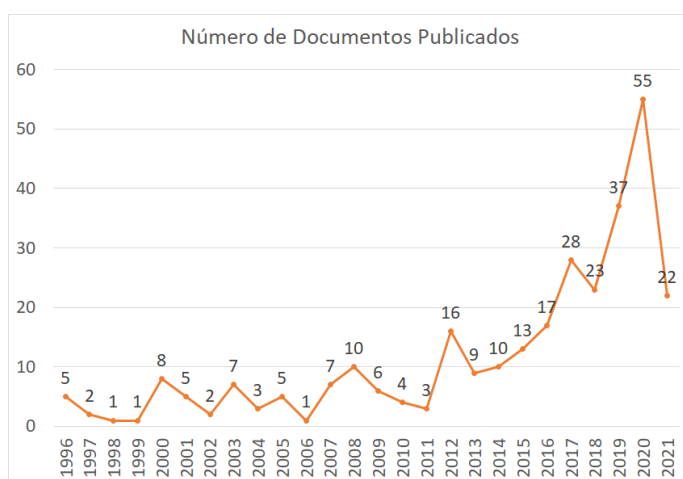


Figura 1 – Número de documentos publicados al año sobre “Precision Farming” en IEEEExplore

Era de interés conocer el tipo de publicación en la que más se concentran documentos en esta base de datos. Esta distribución se presenta en la figura 3. Resulta evidente que la mayoría de las publicaciones que se han realizado sobre el tema estén en conferencias. La cantidad de artículos en Journals es todavía reducida. El interés en esta temática es tan reciente, que aún es bajo el número de documentos publicados tipo divulgación (Magazines). Los libros suelen escribirse cuando los temas se consolidan, y es la razón para que haya o libros publicados en esta temática concreta.

### **3.1. Importancia de las variables medidas**

El conocimiento del comportamiento de las especies en un modelo agrícola particular se puede obtener mediante el monitoreo de su etapa inicial de crecimiento y resulta útil para establecer los rangos de temperatura y nivel de precipitaciones requerido en el correcto desarrollo del cultivo. Para tal fin, el monitoreo debe incluir los eventos bióticos, los factores endógenos y los aspectos filogenéticos, así como sus relaciones con el desarrollo del cultivo. Este modelo permite determinar la forma en que los aspectos climáticos impactan en las condiciones necesarias para favorecer el ciclo de crecimiento de cada uno de los cultivos.

Son de interés las 2 etapas del cultivo correspondientes del cultivo: la etapa vegetativa y la etapa reproductiva. Es decir, desde la germinación de las semillas hasta el surgimiento de la cosecha. Mediante la evaluación de los ciclos productivos de corto y largo plazo, se pueden establecer las correlaciones entre los aspectos fenológicos y las variables climáticas de interés principal: rangos de temperatura, la humedad ambiental relativa y los volúmenes promedio de precipitaciones.

#### **3.1.1. Temperatura**

Se trata de una propiedad física de los cuerpos y sistemas que establece si se hallan en equilibrio térmico o no. La temperatura como concepto proviene de la intención de medir el calor relativo y el frío relativo de los cuerpos, sistemas o fenómenos. También tiene su origen en el hecho de que cuando se proporciona calor a un sistema, este tiende a un aumento de la temperatura en la medida en que no existan procesos de fusión o ebullición. Cuando se tienen sistemas con distinta temperatura, el calor tiende a fluir desde el más caliente (temperatura alta) al más frío (temperatura baja). Es un proceso continuo hasta que se llega al equilibrio térmico.

Desde el punto de vista climático, los flujos de calor se dan por radiación, lo que implica que no se requiere soporte material para transportar el calor de un cuerpo a otro. Resulta de interés el hecho de que, en las estaciones climatológicas, las necesarias en entornos agrícolas para monitorear el clima y poder generar alertas tempranas, la medición de la temperatura se lleva a cabo mediante termómetros secos, así como termómetros para temperaturas máximas y mínimas. Para la obtención de las mediciones fiables y representativas de la zona, se debe proteger a los termómetros de la radiación solar y encontrarse debidamente ventilados. Una forma de protección es la instalación al interior de una carcasa meteorológica que está formada por distintas láminas de madera que proporcionan sombra a los termómetros al tiempo que no obstruyen la ventilación de estos.

### **3.1.2. Humedad**

Esta variable es importante porque cambios mínimos relativos su vapor dará lugar a cambios significativos en el estado del clima. Debido a esto, la determinación de cualquier estado futuro de la atmósfera precisa el análisis de las variaciones de humedad, o lo que es lo mismo, el contenido de vapor de agua en la misma.

La humedad relativa es una variable climatológica que ayuda a determinar qué tan húmedo o seco es el aire en una determinada región. Viene expresado en unidades porcentuales enteras que van desde aire seco absoluto (0%) hasta la saturación (100%). La humedad relativa se suele medir en la atmósfera mediante dos instrumentos: sicrómetro e higrógrafo.

### **3.1.3. Precipitaciones**

Se trata de la lluvia, llovizna, nieve, granizo, entre otras, que se pueden experimentar en una determinada región. La medición de los niveles de precipitación y su correspondiente análisis permite establecer como se distribuirán en el tiempo y en el espacio geográfico. Cuando se miden precipitaciones en una zona, se intenta obtener una muestra representativa de las caídas de lluvias en una región específica. Para realizar esta medición, los instrumentos son el pluviómetro y el pluviógrafo. Detectan y registran las cantidades de precipitación. La unidad de medición de la precipitación es el milímetro (mm). Un mm de lluvia equivale a la caída de un litro de agua en cada uno de los metros cuadrados que conforman un terreno en particular.

## **4. Tecnología de sistema**

### **4.1. ESP32**

Según (Biradar & Shabadi, 2017) se trata de un sistema embebido del tipo SOC (System On Chip), que consiste en un procesador dual core Tensilica Xtensa LX6 el cual trabaja a una frecuencia de reloj de 240 MHz. Además, el sistema cuenta con módulos adicionales para comunicaciones a nivel de la capa de enlace como pueden ser WiFi y Bluetooth. El fabricante de este sistema es la empresa Espressif Systems. El SOC integra toda la tecnología correspondiente a RF (radiofrecuencia) como es una antena, un adaptador de impedancia de RF, un amplificador de potencia de RF, amplificador de bajo ruido (LNA) en la entrada del bloque de comunicaciones del módulo, filtros, bloques para la gestión del consumo de energía. Esta última tecnología hace que el sistema global sea de bajo consumo.

El sistema ESP32 ha sido diseñado para trabajar con tecnología móvil, aplicaciones electrónicas en general e Internet de las Cosas (IoT) en particular. Por eso las funciones de gestión energética son importantes para conseguir consumos ultra bajos. El funcionamiento del módulo incluye ajuste fino del reloj, varios modos de configuración de la potencia y ajuste dinámico del consumo de potencia. En la tabla 1 se resumen las características principales del módulo.

| Característica Principales |                                                       |
|----------------------------|-------------------------------------------------------|
| Procesador principal       | Tensilica Xtensa LX6 de 32 bits                       |
| Wi-Fi                      | 802.11 b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s) |
| Bluetooth                  | v4.2 BR / EDR y Bluetooth Low Energy (BLE)            |
| Frecuencia de Clock        | Programable, hasta 240MHz                             |
| Rendimiento                | hasta 600DMIPS                                        |
| ROM                        | 448KB, para arranque y funciones básicas              |
| SRAM                       | 520KiB, para datos e instrucciones                    |

Tabla 1 – Características principales del ESP32.

Entre las tecnologías posibles a utilizar para servir de nodo en la medición de variables climatológicas, como pueden ser tarjetas Arduino o la Raspberry Pi, la relación prestaciones/precio del módulo ESP32 era lo suficientemente alta como para decantar la decisión a su favor. En la figura 2 se puede ver la distribución de pines I/O del módulo.

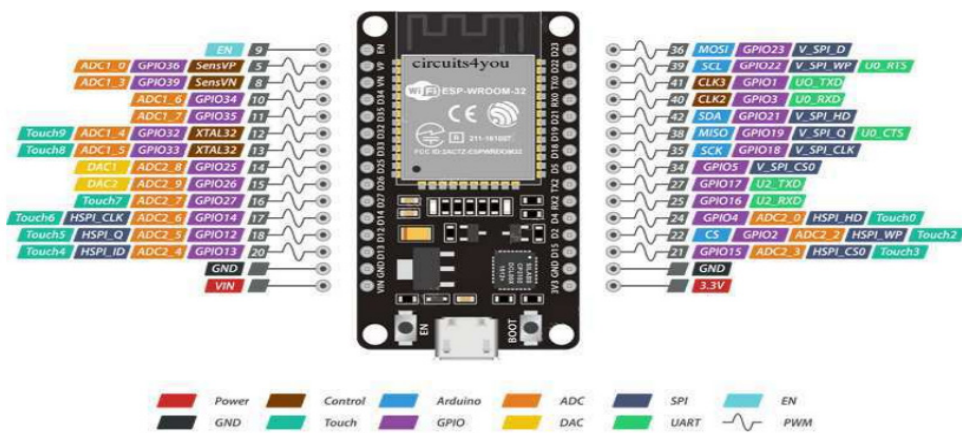


Figura 2 – Distribución de pines I/O del módulo ESP32

Producto de la revisión se lograron encontrar distintos equipos de infraestructura que soportan los datos en ámbitos de agricultura de precisión. Así mismo, los lenguajes de programación que soportan cada una de las tarjetas que permiten la interconexión de los datos.

| Dispositivos                                                                                                    | Procesador                                               | Lenguajes de programación                                                                                                                                                                                                                                                                                | Comunicación/ Control                                 | Almacenamiento |
|-----------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|----------------|
| <b>Tipo 1: RaspberryPi</b><br> | ARM Cortex-A7 CPU 900 MHz Memory 1GB                     | Linux C/C++ and Python APIs libraries                                                                                                                                                                                                                                                                    | WIFI and USB SPI and MQTT protocol GPIO and A/D modul | Micro SD 16 GB |
| <b>Tipo 2: Photon IoT</b><br>  | ARM Cortex-M3 CPU 120 MHz 1MB flash, 128KB RAM           | C language APIs libraries                                                                                                                                                                                                                                                                                | WIFI and USB SPI and MQTT protocol GPIO and A/D ports | -              |
| <b>Tipo 3: SmartPhone</b><br>  | SoC $\mu$ Processor CPU $\geq$ 900 MHz Memory $\geq$ 1GB | Android/iOS Objective C Java libraries                                                                                                                                                                                                                                                                   | WIFI and USB Bluetooth 4G LTE                         | Escalable      |
| <b>Tipo 4: ESP -32</b><br>   | Xtensa Dual-Core 32-bit LX6 con 600 DMIPS                | Esp-idf (Espressif IoT Development Framework) desarrollado por el fabricante del chip, Arduino (en lenguaje C++), Simba Embedded Programming Platform (en lenguaje Python), RTOS's (como Zephyr Project, Mongoose OS, NuttX RTOS), MicroPython, LUA, Javascript (Espruino, Duktape, Mongoose JS), Basic. | HT40 Bluetooth 4.2 y BLE 160 MHz                      | -              |

Tabla 2 – Comparativa entre distintas plataformas tecnológicas para soporte hardware.

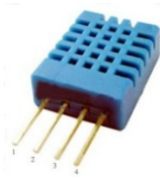
## 4.2. Sensores

### 4.2.1. DHT-11

Se trata de un sensor con la capacidad de medir temperatura y humedad relativa. En las especificaciones del dispositivo se establece que sus rangos de medición son en 20% y 90% para la humedad relativa, con un margen de error de  $\pm 5\%$ , y entre 0 y 50°C para el caso de la temperatura (Klerkx et al., 2019).

Un parámetro importante para los sensores que se utilizan en aplicaciones de tiempo real, en especial si van orientado a sistemas de alertas tempranas, es la rapidez de respuesta o tiempo de respuesta ante variaciones en la temperatura y humedad del ambiente. Para el DHT-11 este tiempo puede variar entre 6 y 30 segundos. La información técnica suministrada por el fabricante destaca algunos aspectos de calidad como la calibración de los sensores, la estabilidad y precisión, así como la característica de que la señal suministrada por el sensor es de tipo digital.

En la Figura 3 se presenta la distribución de pines del sensor. Se recomienda el uso de una resistencia de entre 5k y 10k entre los pines 1 y 2. Una presentación común de este módulo es una tarjeta que lo integra con la mencionada resistencia.



| Pin | Nombre | Descripción                         |
|-----|--------|-------------------------------------|
| 1   | VDD    | Alimentación de energía 3 – 5.5V DC |
| 2   | DATA   | Datos de salida                     |
| 3   | NC     | Sin conexión                        |
| 4   | GND    | Conexión de tierra                  |

Figura 3 – Distribución de pines del sensor DHT-11

El sensor DHT-11 es capaz de capturar muestras en tiempo real, lo cual dependerá de las condiciones del entorno. Por su parte, la temperatura se puede medir y gestionar en diferentes escalas (°C, °F, K, R, entre otros). La humedad relativa es medida en valores porcentuales

### 4.2.2. Pluviómetro RM52203

Se trata de un sensor que utiliza la tecnología de las cazoletas basculantes como principio de medición de la precipitación. La configuración física y el material en el que se fabrican las cazoletas permiten una evacuación total del agua recolectada con el objetivo de minimizar la contaminación ambiental y la ocurrencia de errores en la medición. Para el caso de esta referencia, el área destinada a la recogida de agua es un círculo de 200 cm<sup>2</sup>, y cada giro corresponde a 0.1 mm de precipitación, como es señalado por la organización mundial de meteorología (WMO).

El material de fabricación de este sensor es un termoplástico molido, que presenta una alta resistencia a la corrosión debido al ambiente. Adicionalmente, este pluviómetro cuenta con tornillos para realizar la nivelación de la cazoleta. La salida de agua se puede acoplar a un tubo que permita recoger el total de la precipitación que circula por el sensor con fines de verificación final de la medición. El rango de medida es de 0 – 50 mm/h, con errores de entre  $\pm 2$  y  $\pm 3$ . En la Figura 4 se puede ver la configuración física del pluviómetro analizado.



Figura 4 – Pluviómetro RM52203

### 4.3. Tecnología software

Para poder hacer integraciones tecnológicas vinculadas con el IoT se requieren plataformas que permitan el acopio de datos en tiempo real y su visualización de forma amigable con los usuarios de las soluciones tecnológicas implementadas. En el mercado se pueden encontrar varias de estas plataformas, algunas de las cuales son de código abierto, y otras son propietarias. En algunos casos no es la sólo el software, sino que se puede encontrar todo un ecosistema que incluye también el hardware y todos los elementos necesarios para el desarrollo de la aplicación concreta. Cada una de estas plataformas proporciona unas características particulares, como puede ser el hardware más adecuado, los dominios de aplicación, así como el objetivo general buscado con la plataforma.

Dentro de las muchas que hay en el mercado, en este proyecto se seleccionó Ubidots, al ofrecer una mayor facilidad de integración con dispositivos IoT, tarjetas programables como ESP32 y Arduino. También ofrece una disposición bastante amigable del dashboard para el despliegue de los datos en pantalla. Ubidots es una de esas plataformas de analítica de datos que permite el establecimiento de rangos arbitrarios para las fechas, el cálculo de cifras estadísticas para cualquier período de captura de datos, como la media, la varianza, entre otros. Ubidots ofrece seguridad en todos los procesos de intercambio de datos, mediante el uso de protocolos de cifrado y robustos métodos de autenticación.

## 5. Configuración del sistema

El sensor DHT-11 y el pluviómetro RM52203 se integran a la placa ESP32 mediante la conexión a sus pines de entrada configurados durante la programación del dispositivo.

El ESP32 se configuró de tal manera que utilizara su módulo WiFi para la conexión con un punto de acceso instalado en una sala de un hogar doméstico, toda vez que el prototipo se probó midiendo variables climatológicas en el patio de un hogar ubicado en la ciudad de Barranquilla. El punto de acceso con conexión a Internet tiene la capacidad de enviar los datos a la plataforma Ubidots donde son presentados para que el usuario los observe. El módulo ESP32 se programó mediante el IDE de Arduino, para lo cual fue necesaria la instalación de un complemento en la aplicación, que le permite a la IDE trabajar con el hardware seleccionado. Para llevar a cabo las pruebas se estuvieron enviando los datos medidos por los sensores a la plataforma Ubidots, donde fueron desplegados en un dashboard para que el usuario tenga acceso a ellos. En la Figura 5 se presenta el diagrama general con la integración de las plataformas.



Figura 5 – Integración de las plataformas

### 6. Resultados

La programación en la ESP32 se implementó por medio del entorno de desarrollo de Arduino IDE, el cual tiene una interfaz de desarrollo para ingresar por medio de la consola el ingreso del código fuente que realizar la conexión con servidor en la nube.

```
/* Variables de Temperatura y Humedad a
Ubidots a través del protocolo HTTP..*/
/*****

*Libreria de ubidots para ESP8266
*****/

#include "Ubidots.h"
/*****

* Definir instancias y constantes
*****/

const char* UBIDOTS_TOKEN = "-----"; //Ingresa aquí tu Ubidots TOKEN
```

```

const char* WIFI_SSID = "----"; //Ingresa aquí tu SSID de tu Wi-Fi
const char* WIFI_PASS = "----"; //Ingresa aquí tu contraseña de Wi-Fi
Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);
/*****

* Define instancias del DHT11
*****/

#include <DHT.h> //Cargamos la librería DHT
#define DHTTYPE DHT11 //Definimos el modelo DHT11
#define DHTPIN 2 //Se define el pin para conectar el sensor
DHT dht(DHTPIN, DHTTYPE, 11);
/*****

*Funciones principales
*****/

void setup()
{
  Serial.begin(9600); //Se inicia la comunicación serial
  dht.begin();
  ubidots.wifiConnect(WIFI_SSID, WIFI_PASS);
}

void loop()
{
  float hum = dht.readHumidity(); //Se lee la humedad
  float tem = dht.readTemperature(); //Se lee la temperatura
  //Se imprimen las variables
  Serial.println(hum);
  Serial.println(tem);
  //Añade las variables para ubidots con la lectura de "Temperatura" y "Humedad"
  ubidots.add("Temperatura", tem);
  ubidots.add("Humedad", hum);
  // Enviará datos a una etiqueta de dispositivo que coincida con el ID del dispositivo
  ubidots.send();

```

```
delay(2000);  
}
```

A continuación, en la Figura 6 se presentan algunas de las imágenes obtenidas del dashboard con la información climatológica medida con los sensores seleccionados para el caso de la temperatura y las precipitaciones.



Figura 5 – Dashboard Monitoreo climático

## 7. Conclusión

El tema de la agricultura de precisión, importante en los países en vía de desarrollo para conseguir superar condiciones de pobreza, aumentando su productividad agraria, es un tema de reciente figuración en el campo de la investigación científica y tecnológica. Las primeras publicaciones datan de finales de los años 90s, y sólo superado el año 2010 se evidencio un incremento apreciable de trabajos de investigación relacionados con el tema. Se ha visto a lo largo de esta revisión las necesidades que tiene el sector agrícola de vincular tecnología en la realización de sus actividades, permitiendo una mejor gestión de los procesos productivos del campo, soportando las decisiones de los productores en criterios más objetivos que la mera experiencia, como sucedía hasta hace pocos años. El sistema de monitoreo se logró hacer en un ambiente simulado dentro de la Universidad de la Costa en el Laboratorio de redes del Departamento de Ciencias de la Computación y Electrónica, toda esta implementación termino en la visualización de las variables en tiempo real, lo que permitirá a los usuarios finales saber sobre las condiciones climáticas.

## Referencias

- Abd El-Kader, S. M. (2020). Laying the foundations for an IoT reference architecture for agricultural application domain. *IEEE Access*, 8, 190194-190230. <https://doi.org/10.1109/ACCESS.2020.3031634>
- Aguilar, J., García, R., Toro, M., Pinto, A., & Rodríguez, P. (2020). A systematic literature review on the use of machine learning in precision livestock farming. *Computers and Electronics in Agriculture*, 179. <https://doi.org/10.1016/j.compag.2020.105826>
- Alfred, R., Obit, J. H., Chin, C. P. Y., Havaluddin, H., & Lim, Y. (2021). Towards paddy rice smart farming: A review on big data, machine learning, and rice production tasks. *IEEE Access*, 9. <https://doi.org/10.1109/ACCESS.2021.3069449>
- Ali, J. (2012). Factors affecting the adoption of information and communication technologies (ICTs) for farming decisions. *Journal of Agricultural and Food Information*, 13(1), 78–96. <https://doi.org/10.1080/10496505.2012.636980>
- Amanullah, M. A., Habeeb, R. A. A., Nasaruddin, F. H., Gani, A., Ahmed, E., Nainar, A. S. M., Akim, N. M., & Imran, M. (2020). Deep learning and big data technologies for IoT security. *Computer Communications*, 151. <https://doi.org/10.1016/j.comcom.2020.01.016>
- Arsovski, Z., Lula, P., & Đorđević, A. (2016). Impact of ICT on quality of life.
- Bankole, F. O., Shirazi, F., & Brown, I. (2011). Investigating the impact of ICT investments on human development. In *EJISDC* (Vol. 48). <http://www.ejisdc.org>
- Bhatia, G., Joshi, N., Iyengar, S., Rajpal, S., & Mahadevan, K. (2021). Crop Prediction Based on Environmental Conditions and Disease Prediction. *Smart Innovation, Systems and Technologies*, 195. [https://doi.org/10.1007/978-981-15-7078-0\\_31](https://doi.org/10.1007/978-981-15-7078-0_31)
- Biradar, H. B., & Shabadi, L. (2017). Review on IOT based multidisciplinary models for smart farming. *RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings*, 2018-Janua, 1923–1926. <https://doi.org/10.1109/RTEICT.2017.8256932>
- Bouachir, W., Ihou, K. E., Gueziri, H. E., Bouguila, N., & Belanger, N. (2019). Computer Vision System for Automatic Counting of Planting Microsites Using UAV Imagery. *IEEE Access*, 7. <https://doi.org/10.1109/ACCESS.2019.2923765>
- Bula, A. (2020). Importancia de la Agricultura en el Desarrollo.
- Cecinati, F., Moreno-Ródenas, A. M., Rico-Ramirez, M. A., ten Veldhuis, M. C., & Langeveld, J. G. (2018). Considering rain gauge uncertainty using kriging for uncertain data. *Atmosphere*, 9(11). <https://doi.org/10.3390/atmos9110446>
- de la Casa, A., Ovando, G., Bressanini, L., Martínez, J., Díaz, G., & Miranda, C. (2018). Soybean crop coverage estimation from NDVI images with different spatial resolution to evaluate yield variability in a plot. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146. <https://doi.org/10.1016/j.isprsjprs.2018.10.018>

- Dobrescu, R., Merezeanu, D., & Mocanu, S. (2019). Context-aware control and monitoring system with IoT and cloud support. *Computers and Electronics in Agriculture*, 160. <https://doi.org/10.1016/j.compag.2019.03.005>
- Gašparović, S. (2019). Impact of ICT on some segments of everyday life of highschool population of the city of Zagreb. *IJRAR- International Journal of Research and Analytical Reviews*, 6(2), 62–65.
- Ge, X. Y., Ding, J. L., Wang, J. Z., Sun, H. L., & Zhu, Z. Q. (2020). A New Method for Predicting Soil Moisture Based on UAV Hyperspectral Image. *Guang Pu Xue Yu Guang Pu Fen Xi/Spectroscopy and Spectral Analysis*, 40(2). [https://doi.org/10.3964/j.issn.1000-0593\(2020\)02-0602-08](https://doi.org/10.3964/j.issn.1000-0593(2020)02-0602-08)
- Gopinath, R. (2023). Perception of ICT in farming practices with special reference to e-commerce in agriculture. *IJRAR-International Journal of Research and Analytical Reviews*, 6. <http://ijrar.com/>
- Hernandez, R. M. (2017). Impact of ICT on education: Challenges and perspectives. *Propósitos y Representaciones*, 5(1), 325–347. <https://doi.org/10.20511/pyr2017.v5n1.149>
- Khan, M. A., & Salah, K. (2018). IoT security: Review, blockchain solutions, and open challenges. *Future Generation Computer Systems*, 82. <https://doi.org/10.1016/j.future.2017.11.022>
- Klerkx, L., Jakku, E., & Labarthe, P. (2019). A review of social science on digital agriculture, smart farming and agriculture 4.0: New contributions and a future research agenda. *NJAS - Wageningen Journal of Life Sciences*, 90–91(October), 100315. <https://doi.org/10.1016/j.njas.2019.100315>
- Kocian, A., & Incrocci, L. (2020). Learning from Data to Optimize Control in Precision Farming. *Stats*, 3(3). <https://doi.org/10.3390/stats3030018>
- Marwa, M. E., Mburu, J., Oburu, R. E. J., Mwai, O., & Kahumbu, S. (2020). Impact of ICT based extension services on dairy production and household welfare: The case of iCow service in Kenya. *Journal of Agricultural Science*, 12(3), 141. <https://doi.org/10.5539/jas.v12n3p141>
- Miller, G. A., Hyslop, J. J., Barclay, D., Edwards, A., Thomson, W., & Duthie, C. A. (2019). Using 3D Imaging and Machine Learning to Predict Liveweight and Carcass Characteristics of Live Finishing Beef Cattle. *Frontiers in Sustainable Food Systems*, 3. <https://doi.org/10.3389/fsufs.2019.00030>
- Minagricultura. (2018). *Un Campo para la Equidad: Política Agropecuaria y de Desarrollo Rural*.
- Mohammad El-Basioni, B. M., & Abd El-Kader, S. M. (2020). Laying the foundations for an IoT reference architecture for agricultural application domain. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2020.3031634>

- Mohamad Noor, M. B., & Hassan, W. H. (2019). Current research on Internet of Things (IoT) security: A survey. *Computer Networks*, 148. <https://doi.org/10.1016/j.comnet.2018.11.025>
- Newase, A. D., Sheetlani, J., Sai, S., & Patil, R. D. (2017). A literature review on impact of information and communication technology tools on rural society of India. *Indian Journal of Computer Science and Engineering*, 8(3), 235–240.
- Oussous, A., Benjelloun, F. Z., Ait Lahcen, A., & Belfkih, S. (2018). Big Data technologies: A survey. In *Journal of King Saud University - Computer and Information Sciences*, 30, (4). <https://doi.org/10.1016/j.jksuci.2017.06.001>
- Panesar, G. S., Venkatesh, D., Rakhra, M., Jairath, K., & Shabaz, M. (2021). Agile software and business development using artificial intelligence. *Annals of the Romanian Society for Cell Biology*, 25(2).
- Parra-Peña, R., Puyana, R., & Yepes-Chica, F. (2021). Análisis de la productividad del sector agropecuario en Colombia y su impacto en temas como: encadenamientos productivos, sostenibilidad e internacionalización, en el marco del programa Colombia más competitiva.
- Punchihewa, D. J., & Wimalaratne, P. (2010). Towards an ICT enabled farming community. *E-Governance in Practice, India* (pp. 201–207). <https://www.icta.lk>
- Sidhu, S., Mohd, B. J., & Hayajneh, T. (2019). Hardware security in IoT devices with emphasis on hardware trojans. *Journal of Sensor and Actuator Networks*, 8(3). <https://doi.org/10.3390/jsan8030042>
- Thangaraju, G., Agnes, X., & Rani, K. (2016). A study to improve the organic farming and the impact of ICT in organic agriculture with special reference to Perambalur district in Tamilnadu (Data analysis with K-means clustering algorithm in datamining). *International Journal of Emerging Technology in Computer Science & Electronics*, 20.
- UNGRD. (2021). Sistema de Alerta Temprana. Retrieved from <http://portal.gestiondelriesgo.gov.co/Paginas/SAT.aspx>
- Unal, Z. (2020). Smart farming becomes even smarter with deep learning - A bibliographical analysis. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2020.3000175>
- Vergara-Díaz, O., Kefauver, S., Araus, J. L., & Aranjuelo, I. (2020). Development of novel technological approaches for a reliable crop characterization under changing environmental conditions. *NIR News*, 31(7–8). <https://doi.org/10.1177/096033602097874>
- Wei, H. E., Grafton, M., Bretherton, M., Irwin, M., & Sandoval, E. (2021). Evaluation of point hyperspectral reflectance and multivariate regression models for grapevine water status estimation. *Remote Sensing*, 13(16). <https://doi.org/10.3390/rs13163198>

# Propuesta de unificación de herramientas bioinformáticas para el análisis de secuencias proteómicas y genómicas de algas y hongos

Victor Terrón-Macias<sup>1</sup>, Jezreel Mejía<sup>1</sup>

{victor.terron, jmejia}@cimat.mx

<sup>1</sup> Centro de Investigación en Matemáticas, Parque Quantum, Ciudad del Conocimiento, Av. Lassec, Andador Galileo Galilei, Manzana 3, Lote 7, CP 98160 Zacatecas, Zac., México.

DOI: 10.17013/risti.53.122–137

**Resumen:** El análisis de secuencias biológicas es esencial en bioinformática, pero la diversidad de herramientas disponibles presenta desafíos significativos. Este artículo propone una aplicación web que unifica diferentes funcionalidades útiles en el análisis de secuencias biológicas. Desarrollada en Python y Django, integra funcionalidades de BioPython, BioPandas, 3Dmol.js y BLAST, entre otras, que además posibilita su uso a usuarios sin conocimientos de programación. La herramienta abarca el análisis de secuencias de ADN y proteínas, transcripción, alineación, búsqueda de secuencias y más. Dentro de las características más destacables está la eficiencia, precisión al obtener datos cuantitativos del proceso que se está ejecutando y la facilidad de uso.

**Palabras-clave:** bio-informática, bio-tecnología, análisis de secuencias biológicas, sin código.

## *Proposal for the unification of bioinformatics tools for the analysis of proteomic and genomic sequences of algae and fungi*

**Abstract:** Biological sequence analysis is essential in bioinformatics, but the diversity of available tools presents significant challenges. This article proposes a web application that unifies useful functionalities in biological sequence analysis. Developed in Python and Django, it integrates features from BioPython, BioPandas, 3Dmol.js, and BLAST, among others, making it usable for users without programming knowledge. The tool covers the analysis of DNA and protein sequences, transcription, alignment, sequence search, and more. Its most notable features are efficiency, precision in obtaining quantitative data from the running process, and ease of use.

**Keywords:** bioinformatics, biotechnology, biological sequence analysis, no code.

## 1. Introducción

El análisis de secuencias biológicas es una de las actividades fundamentales en bioinformática (Dimitrios Vasileiou, 2023) facilita la genómica estructural, comparativa y funcional, sirviendo como marco subyacente para predicciones sistemáticas de estructura/función, clasificación y anotación de proteínas. Este análisis ayuda a identificar funciones o características cruciales de los organismos (Katie Emelianova, 2023) (Azza E. Ahmed, 2021), como la complejidad de la regulación transcripcional y la información molecular, entre otras características o información valiosa.

Ante la gran cantidad de datos biológicos, es imperativo recurrir a software eficiente y probado que facilite estos análisis, más aún cuando se ejecutan a gran escala porque algunas de las herramientas actuales son complejas de utilizar (Azza E. Ahmed, 2021).

En este contexto, se ha identificado que existe una amplia variedad de software bioinformático como BioPython (Cock, 2009), cogent3 (Cogent3 Team), cactus (Joel Armstrong, 2020), PyCogent (Rob Knight, 2007), PyEvolve (Andrew Butterfield, 2004), entre otros; que realizan tareas específicas, y que a su vez promueven la dispersión de herramientas. Como resultado, profesionales relacionados al campo encuentran ineficiencias en sus tareas diarias al utilizar varios softwares en lugar de uno.

Jean-Baptiste Cazier en 2013 realizó una encuesta en donde obtuvo 47 respuestas de expertos en el campo (Cazier, 2013), en donde plantea la misma problemática sobre la existencia de muchas herramientas bioinformáticas, como resultado identificó los siguientes puntos:

1. 89% de las respuestas coinciden en que hay demasiadas herramientas bioinformáticas.
2. 34% de las opiniones coinciden en que es difícil encontrar software adecuado para tareas específicas.
3. 30% mencionan que es necesario compartir experiencias y conocimientos para hacer un mejor uso de las herramientas disponibles.
4. 13% de las opiniones indicaron que es necesario hacer un esfuerzo conjunto por unificar las herramientas en un solo entorno.

Otro aspecto relevante que se obtuvo de esta encuesta es que los expertos encuentran muchas dificultades en el uso de las herramientas bioinformáticas existentes, esto se presenta en los siguientes puntos:

1. 25% de las opiniones mencionan dificultad para encontrar software adecuado para tareas específicas.
2. 13% mencionan la falta de un esfuerzo conjunto para unificar las herramientas en un solo entorno.
3. 15% indicaron la complejidad y la curva de aprendizaje asociadas con algunas herramientas ante la necesidad de tener conocimientos de programación o de compilación de archivos fuente.
4. 11% resaltaron los problemas de documentación y falta de comprensión de los algoritmos subyacentes en algunas herramientas.
5. 15% mencionan la proliferación de herramientas que genera la confusión sobre cuál utilizar.

6. 17% indican la necesidad de actualización y mantenimiento continuo de las herramientas.
7. 9% mencionan la falta de recursos para evaluar y comparar las herramientas disponibles.
8. 15% indicaron las dificultades en la instalación y configuración de algunas herramientas.
9. 15% de los expertos coinciden en que existe variabilidad en los resultados obtenidos con diferentes herramientas para la misma tarea.

Ante este contexto este artículo presenta una propuesta de aplicación web con el objetivo de unificar diferentes herramientas de análisis en un mismo entorno enfocado en el análisis elemental y sin código que conjunte en un solo entorno diferentes herramientas atendiendo las problemáticas identificadas.

## **2. Plataformas relacionadas**

Para identificar las funcionalidades que se pueden implementar dentro de la propuesta de unificación de herramientas se realizó una investigación explorando las principales plataformas científicas relacionadas al campo que incluyen diferentes herramientas, algunas de las principales plataformas identificadas son:

1. Expasy.
2. NCBI Tools.
3. Chimera.
4. BLAST.

En la siguiente subsección se presentan las funciones que cubre cada una de estas plataformas.

### **2.1. Expasy**

Ofrece una amplia variedad de funciones relacionadas con el análisis y la manipulación de datos biológicos, dentro de las cuales están:

1. Análisis de Secuencias: Capacidad para analizar secuencias biológicas.
2. Predicción de Estructuras: Previsión de la estructura tridimensional de proteínas y ácidos nucleicos.
3. Búsqueda de Homologías: Identificación de secuencias similares en bases de datos.
4. Diseño de Primers: Herramientas para el diseño eficiente de cebadores para PCR.
5. Análisis de Expresión Genética: Evaluación de la expresión génica a nivel transcriptómico.
6. Anotación Funcional: Asignación de funciones biológicas a secuencias.
7. Bases de Datos: Acceso a una variedad de bases de datos biológicas.

### **2.2. NCBI Tools**

La suite de herramientas de NCBI ofrece una amplia gama de recursos para la investigación biomédica:

- 1. Bases de Datos Biomédicas: Repositorios de datos biomédicos para investigación y análisis.
- 2. BLAST: Herramienta para comparar secuencias biológicas con bases de datos.
- 3. Entrez: Sistema de búsqueda y recuperación de información biomédica.
- 4. Información de Genes: Recursos para acceder y comprender información genética.
- 5. Exploración de Genomas y Especies: Herramientas para explorar información genómica.

2.3. Chimera

Chimera se centra en la visualización y modelado de estructuras moleculares:

- 1. Visualización de Estructuras Moleculares: Representación visual de estructuras tridimensionales.
- 2. Modelado de Estructuras: Creación y manipulación de modelos tridimensionales de macromoléculas.
- 3. Análisis de Interacciones: Evaluación de interacciones moleculares.
- 4. Renderización de Imágenes: Generación de imágenes de alta calidad para la presentación de datos.
- 5. Análisis de Archivos PDB: Exploración y análisis de datos estructurales en formato PDB.

2.4. BLAST

BLAST es una herramienta ampliamente utilizada para comparar secuencias biológicas:

- 1. Comparación de Secuencias: Identificación de similitudes entre secuencias biológicas.
- 2. Búsqueda Rápida en Bases de Datos: Acceso rápido a información relevante en bases de datos.
- 3. Alineación de Secuencias: Alineación de secuencias para identificar regiones conservadas.
- 4. Variantes y Mutaciones: Análisis de variaciones genéticas y mutacionales.
- 5. BLASTp, BLASTn, etc.: Variantes de BLAST para diferentes tipos de secuencias.

Considerando las funcionalidades identificadas en cada plataforma y con el objetivo de unificar la mayor cantidad de funcionalidades posibles, se realizó una comparativa entre ellas y la funcionalidad que cubren (ver Tabla 1).

| Funcionalidad                  | Expasy | NCBI Tools | Chimera | BLAST |
|--------------------------------|--------|------------|---------|-------|
| Análisis de secuencias         | ✓      | ✓          |         |       |
| Predicción de estructuras      | ✓      |            | ✓       |       |
| Búsqueda de homologías         | ✓      |            |         |       |
| Diseño de primers              | ✓      |            |         |       |
| Análisis de expresión genética | ✓      |            |         |       |
| Anotación funcional            | ✓      |            |         |       |
| Bases de datos                 | ✓      | ✓          |         |       |

| Funcionalidad                                       | Expasy | NCBI Tools | Chimera | BLAST |
|-----------------------------------------------------|--------|------------|---------|-------|
| Cálculo y generación de árboles filogenéticos       |        |            | ✓       |       |
| Análisis de archivos PDB y visor de PDB             |        |            | ✓       |       |
| Traducción y transcripción de secuencias            |        |            |         |       |
| BLAST (BLAST-P, BLAST-N, BLAST-X, TBLASTN, TBLASTX) |        | ✓          |         | ✓     |
| Entrez                                              |        |            |         |       |
| Información de genes                                |        | ✓          |         |       |
| Archivos de artículos científicos                   |        | ✓          |         |       |
| Exploración de genomas y especies                   |        | ✓          |         |       |
| Transcripción inversa                               |        | ✓          |         |       |
| Análisis de Propiedades de Proteomas y ADN          |        | ✓          |         |       |
| Alineación y traducción de secuencias               |        |            |         | ✓     |
| Análisis de interacciones                           |        |            | ✓       |       |
| Renderización de imágenes                           |        |            | ✓       |       |
| Variantes y mutaciones                              |        |            |         | ✓     |

Tabla 1 – Comparativa respecto a la funcionalidad cubierta de cada plataforma.

Con base a la anterior comparativa, se identifica que del total de funcionalidades identificadas la plataforma que cubre mayor cantidad de esas funciones es NCBI Tools con un 43% de cobertura. De acuerdo al resultado de la Tabla 1, se establecieron las funcionalidades que debería tener la herramienta web que permitiría unificar diversas funcionalidades estableciendo que la plataforma propuesta cubrirá con 52% del total de las funcionalidades, tal como se ve en la Tabla 2, los signos de guión significan que la propuesta cubre parcialmente con esa funcionalidad:

| Funcionalidad                                 | Expasy | NCBI Tools | Chimera | BLAST | Propuesta |
|-----------------------------------------------|--------|------------|---------|-------|-----------|
| Análisis de secuencias                        | ✓      | ✓          |         |       | ✓         |
| Predicción de estructuras                     | ✓      |            | ✓       |       |           |
| Búsqueda de homologías                        | ✓      |            |         |       |           |
| Diseño de primers                             | ✓      |            |         |       | —         |
| Análisis de expresión genética                | ✓      |            |         |       | ✓         |
| Anotación funcional                           | ✓      |            |         |       |           |
| Bases de datos                                | ✓      | ✓          |         |       | ✓         |
| Cálculo y generación de árboles filogenéticos |        |            | ✓       |       | ✓         |
| Análisis de archivos PDB y visor de PDB       |        |            | ✓       |       | ✓         |
| Traducción y transcripción de secuencias      |        |            |         |       | ✓         |

| Funcionalidad                              | Expasy | NCBI Tools | Chimera | BLAST | Propuesta |
|--------------------------------------------|--------|------------|---------|-------|-----------|
| BLAST                                      |        | ✓          |         | ✓     | ✓         |
| Entrez                                     |        | ✓          |         |       | —         |
| Información de genes                       |        | ✓          |         |       | ✓         |
| Archivos de artículos científicos          |        | ✓          |         |       |           |
| Exploración de genomas y especies          |        | ✓          |         |       |           |
| Transcripción inversa                      |        | ✓          |         |       | ✓         |
| Análisis de Propiedades de Proteomas y ADN |        | ✓          |         |       | ✓         |
| Alineación y traducción de secuencias      |        |            |         | ✓     | ✓         |
| Análisis de interacciones                  |        |            | ✓       |       |           |
| Renderización de imágenes                  |        |            | ✓       |       |           |
| Variantes y mutaciones                     |        |            |         | ✓     |           |

Tabla 2 – Cumplimiento de funcionalidades de la propuesta de plataforma.

La siguiente sección aborda la propuesta de unificación, así como la selección de tecnologías que ayudan a lograr unificar las funcionalidades en un solo entorno sin necesidad de que el usuario tenga conocimientos de programación o de compilación de archivos.

3. Propuesta de unificación

Se consideró un conjunto completo de software y herramientas bioinformáticas ya existentes que cuentan con una amplia gama de herramientas que facilitan el análisis de secuencias biológicas, como ExpASy (Gasteiger, 2003) y NCBI Services (Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information, 2017), con el objetivo de identificar las funciones que se pudieran unificar en un mismo entorno.

Esta propuesta de herramienta fue desarrollada utilizando Python como lenguaje de programación, misma que integra librerías que han sido probadas como: BioPython (Cock, 2009), BioPandas (Raschka, 2017) y 3Dmol (Rego, 2014). Mismas que fueron seleccionadas por su estabilidad, confiabilidad y funcionamiento extensivo en el análisis de secuencias biológicas.

3.1. Arquitectura de la aplicación

La arquitectura de la aplicación está estructurada para optimizar la integración de varias funcionalidades de diferentes librerías de Python bajo el framework DJANGO, por ello esta propuesta reúne herramientas que se enfocan en los siguientes puntos:

- 1. Análisis de secuencias de ADN (ver Figura 1, con la secuencia de entrada “ACGTAAACGT”).
  - a. Análisis de propiedades: recibe como entrada una secuencia de nucleótidos, como resultado entrega: secuencia complementaria, secuencia

complementaria inversa, distribución de los aminoácidos presentes con su gráfico.

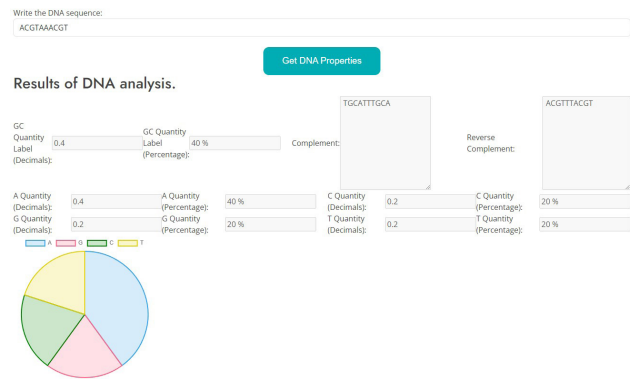


Figura 1 – Análisis de secuencias de ADN.

- i. Como resultado nos entrega una cantidad de GC de 0.4, equivalente al 40% porcentualmente.
  - ii. Adicionalmente nos entrega su secuencia complementaria “TGCATTTGCA” y su secuencia reversa complementaria “ACGTTTACGT”.
  - iii. Finalmente nos entrega la distribución de los elementos de la secuencia en decimales y porcentualmente A 40%,GCT con una distribución del 20% cada uno.
2. Análisis de secuencias proteómicas (ver Figura 2, con la secuencia de entrada *MEQPATKRRKQHNNLETEPTGADYERMSAWLQREGADFANVTIHKSHESSEGYGIYAARA*).

| Results of PROTEOME analysis. |                                   |                                                |
|-------------------------------|-----------------------------------|------------------------------------------------|
| #                             | Parameter                         | Value                                          |
| 1                             | Length of the Proteome:           | 60                                             |
| 2                             | Molecular weight:                 | 6878.4238000000005                             |
| 3                             | Aromaticity:                      | 0.08333333333333333                            |
| 4                             | Instability index:                | 64.68499999999997                              |
| 5                             | Iso-electric point:               | 5.849550819396971                              |
| 6                             | Secondary Structure Fraction:     | (0.16666666666666666, 0.2, 0.3333333333333333) |
| 7                             | Molar Extinction Coefficient (E): | (9970, 9970)                                   |
| 8                             | E Reduced cysteines:              | 9970                                           |
| 9                             | Disulfide bridges:                | 9970                                           |
| 10                            | Gravy:                            | -1.2133333333333336                            |
| 11                            | Charge at pH:                     | 11.926312289610369                             |

Figura 2 – Análisis de propiedades.

- a. Análisis de propiedades: recibe como entrada una secuencia de proteínas y un pH, como resultado entrega: tamaño de la secuencia, peso molecular, aromaticidad, índice de inestabilidad, punto isoeléctrico, estructura secundaria, coeficiente de extinción molar, numero de puentes de disulfuro, índice de hidropaticidad, carga eléctrica e hidrofobicidad.
  - i. Como resultados entrega la longitud de la secuencia: 60, peso molecular: 6878.4238 g/mol, aromaticidad: 0.083, índice de inestabilidad: 64.68, entre otros.
3. Transcripción y transcripción inversa (ver Figura 3, con la secuencia de entrada de GTGAAAAAGATGCAATCTATCGTACTCGCACTTTCCCTGGTTC TGGTCGCTCCCATGGCAGCACAGGCTGCGGAAATTACGTTAGTCCCGTC AGTAAAATTACAGATAGGCGATCGTGATAATCGTGGCTATTACT GGGATGGAGGTCCTGCGCGACCGGCTGGTGGAAACAACAT GAGGCAATCGCTGGCACCTACACGGACCGCCGCCACCGCCGCGCCAC CATAAGAAAGCTCCTCATGATCATCACGGCGGTCATGGTCCAGGCAAACAT CACCGCTAA).
- a. Transcripción: recibe como entrada una secuencia de nucleótidos, como salidas entrega la transcripción de la secuencia a mRNA y hebras plantilla.
  - b. Transcripción inversa: recibe como entrada una secuencia de mRNA y como salida entrega la secuencia de nucleótidos.

**Important:** Only Coding DNA.

Write the DNA sequence:

GTGAAAAAGATGCAATCTATCGTACTCGCACTTTCCCTGGTTCGCTGCTCCCATGGCAGCACAGGCTGCGGAAATTACGTTAGTCCCGTCAGTAAATTACAGATAGGCGATCGTATAATCG

Transcript DNA sequence

**DNA Template Strand**

5' to 3':

TTAGCGGTGATGTTTGCCTGGACCATGACCGCGTGATGATCATGAGGAGCTTCTTATGGTGGCGCGGCGGTGGCGGCGGTCGGTGTAGGTGCCAGCGATTGCCTCGCCATTCAATGTTGTTCCACGACCGG

3' to 5':

GTGAAAAAGATGCAATCTATCGTACTCGCACTTTCCCTGGTTCGCTGCTCCCATGGCAGCACAGGCTGCGGAAATTACGTTAGTCCCGTCAGTAAATTACAGATAGGCGATCGTATAATCGTGGCTATTACTCG

Figura 3 – Transcripción y transcripción inversa.

- i. En la figura se aprecia el resultado, nos entrega la cadena completa de 5' a 3' y de 3' a 5'.
4. Traducción de secuencias (ver Figura 4, con la secuencia de entrada AUUU UCUUUGCUCUUGAGCUCUGGCACUUCUCUGCUGCUGUCUG).
- a. Como entradas recibe secuencias de mRNA, selecciona la tabla de codón a utilizar y como salida entrega el tipo de secuencia y la traducción de mRNA a ADN de la secuencia.

Write the mRNA or DNA sequence:

AUUUUCUAUAGCUCUGGACGUCUGGCACUUCUCGUCGUCUGCG

Select the codon table that you want to use to translate the sequence:

Standard Code.

☐ Vertebrate Mitochondrial Code.

☐ Yeast Mitochondrial Code.

☐ The Mold, Protozoan, and Coelenterate Mitochondrial Code and the Mycoplasma/Spiroplasma Code.

☐ The Invertebrate Mitochondrial Code.

☐ The Ciliates, Dasyatis/Dacodon and Hemitoma Nuclear Code.

☐ The Echinoderm and Flatworm Mitochondrial Code.

☐ The Bacterial, Archaeal and Plant Plastid Code.

☐ The Alternative Yeast Mitochondrial Code.

☐ The Ascidian Mitochondrial Code.

☐ The Alternative Flatworm Mitochondrial Code.

☐ Baphariina Mitochondrial Code.

Chlorophycean Mitochondrial Code.

Trematode Mitochondrial Code.

Scolecemoides obliquus Mitochondrial Code.

Thraustochytrium Mitochondrial Code.

Rhabdopleurina Mitochondrial Code.

Candidate Division SR1 and Gracilibacteria Code.

Pachysolen lamnopsis Nuclear Code.

Karyorelict Nuclear Code.

Cyclorostomus Nuclear Code.

Mesonotum Nuclear Code.

Petritrich Nuclear Code.

Blasticotheria Nuclear Code.

Cephaloscyphidia Mitochondrial UAA-Tyr Code.

Try with all (Brute Force).

Translate sequence

This is a mRNA sequence  
IFFAELWHSAAV  
1- Standard Code.

Figura 4 – Traducción de la secuencia.

- i. Como resultados nos entrega, el tipo de secuencia de entrada, el resultado de la traducción de la secuencia que es: “IFFALELWHFSAAV” y la tabla de codones utilizada para la traducción: tabla de codón estándar.
5. Alineación de secuencias por pares (ver Figura 5, con secuencia de entrada 1: “ATTTTCTTTGCTCTTGAGCTCTGGCACTTCTCTGCTGCTGTCTG” y secuencia de entrada 2: “ATTTTCTTGCTCTTGAGCTCTGGCACTTCTCTG CTGCTGTC”).
  - a. El usuario ingresa dos secuencias que quiera alinear, seleccionando también el modo de alineación, si es global o local y como salida la herramienta entrega estadísticas del proceso de alineación, así como las secuencias alineadas.

## Sequence Aligner Tool

Write the first sequence:

ATTTCCTTGCTGCTGAGCTGCGACCTCTGCTGCTGCTGCTG

Write the second sequence:

ATTTCCTTGCTGCTGAGCTGCGACCTCTGCTGCTGCTGCTG

Select the mode of alignment:

☒ Global Alignment.  
☐ Local Alignment.

Match Score:

Mismatch Score:

Gap Score:

Select the penalization matrix:

☒ BLOSUM62.  
☐ BENNER2.  
☐ BENNER6.  
☐ BLASTN.  
☐ BLASTP.  
☐ BLOSUM45.  
☐ BLOSUM62.  
☐ BLOSUM80.  
☐ BLOSUM90.  
☐ DAYHOFF.  
☐ FENG.  
☐ GENETIC.  
☐ GONNET1992.  
☐ HOXD70.  
☐ JOHNSON.  
☒ NONE.

Align sequences

Parameters of aligner:

Pairwise sequence aligner with parameters wildcard: Home match\_score: 1.000000 mismatch\_score: 0.000000 target\_internal\_open\_gap\_score: 0.000000 target\_internal\_extend\_gap\_score: 0.000000 target\_left\_open\_gap\_score: 0.000000 target\_left\_extend\_gap\_score: 0.000000

Figura 5 – Alineación de secuencias por pares.

- i. Como resultados nos entrega la secuencia alineada y las estadísticas del alineador (puntaje de coincidencia, puntaje de no coincidencia, tipo de alineamiento, entre otros).

## 6. Búsqueda de secuencias.

- a. El usuario proporciona una secuencia que puede ser proteína, nucleótido o gen, la aplicación hace una búsqueda utilizando BLAST y como salida entrega una lista de organismos que comparten similitudes con la secuencia. De esta funcionalidad no se coloca imagen por la extensión de esta.

## 7. Visualizador de archivos PDB (ver Figura 6, como valores de entrada recibe un archivo PDB que puede ser descargado en cualquier portal que ofrezca este tipo de visualizaciones de las proteínas de los organismos).

- a. La herramienta recibe como entrada un archivo PDB, y como salida entrega la estructura 3D de la proteína con sus diferentes modos de visualización.

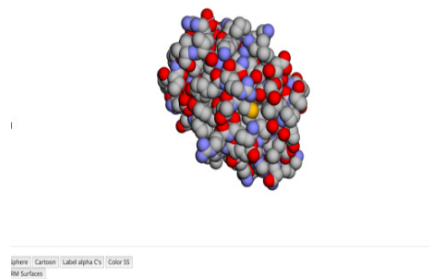


Figura 6 – Visualizador de archivos PDB.

- i. Como resultados entrega la visualización, así mismo cuenta con varios modos de visualización (palos, caricatura, esferas, vectores, etc.) dentro de los cuales el usuario puede interactuar para cambiar el modo de visualización.

## 8. Árboles filogenéticos (ver Figura 7).

- a. La herramienta recibe como entrada un archivo en formato CLUSTAL, como salidas entrega el cálculo de distancias entre las secuencias biológicas y genera un árbol filogenético.

### Distance Calculations.

|      | Seq1     | Seq2     | Seq3     | Seq4     | Seq5     |
|------|----------|----------|----------|----------|----------|
| Seq4 | 0.192308 | 0.000000 |          |          |          |
| Seq0 | 0.230769 | 0.269231 | 0.000000 |          |          |
| Seq3 | 0.230769 | 0.192308 | 0.230769 | 0.000000 |          |
| Seq2 | 0.230769 | 0.192308 | 0.230769 | 0.152384 | 0.000000 |
| Seq5 | 0.230769 | 0.230769 | 0.192308 | 0.192308 | 0.000000 |

```

Tree(rooted=False)
Clade(branch_length=0, name="Inner4")
Clade(branch_length=0.014423876923876886, name="Inner3")
Clade(branch_length=0.024838461538461536, name="Inner1")
Clade(branch_length=0.10096153846153844, name="Seq1")
Clade(branch_length=0.09134615384615385, name="Seq4")
Clade(branch_length=0.024838461538461536, name="Inner2")
Clade(branch_length=0.0081828651528651, name="Seq5")
Clade(branch_length=0.1121794871794872, name="Seq0")
Clade(branch_length=0.07692387692387694, name="Seq3")
Clade(branch_length=0.07692387692387693, name="Seq2")

```

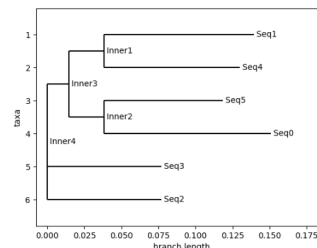


Figura 7 – Árboles filogenéticos.

- i. Como resultado, entrega el cálculo de distancias entre cada rama de las secuencias del archivo CLUSTAL, un diagrama que permite visualizar las distancias de las secuencias y el texto que forma cada rama del diagrama.
9. Análisis de MOTIFS y expresiones regulares (ver Figura 8, con las secuencias de entrada: TACAA, TACGC, TACAC, TACCC, AACCC, AATGC, AATGC).
- a. Análisis de MOTIFS: El usuario recibe una lista de MOTIFS, como salidas entrega una matriz de conteo, la secuencia de consenso, secuencia de consenso degenerado, reversa complementaria y la secuencia de consenso complementaria degenerada.
  - b. Expresiones regulares: la herramienta recibe una expresión regular de lo que va a buscar globalmente en la base de datos que se le cargue.

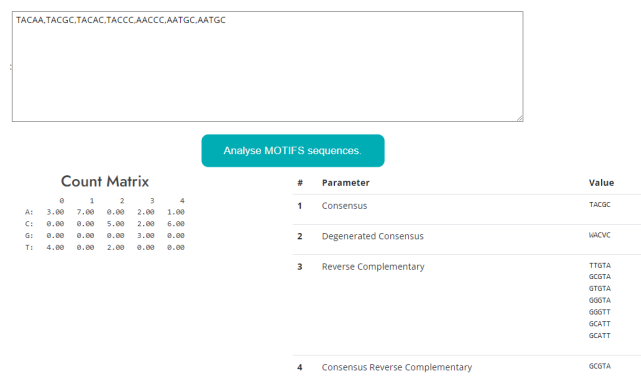


Figura 8 – Análisis de MOTIFS.

- i. Como resultados nos entrega la matriz de conteo, la secuencia consenso, la secuencia de consenso degenerado, la secuencia reversa complementaria y la secuencia consenso reversa complementaria.
10. Conexión con base de datos de hongos y algas con datos provenientes del JGI (JGI, 2022) o de cualquier otra fuente.
- a. El sistema se conecta a una base de datos MONGODB que debe ser instalada previamente, misma que contiene todas las secuencias que el usuario decida cargar ya sea de hongos, algas o del tipo de organismo que el usuario trabaje.

En las subsecciones siguientes se aborda la interfaz de usuario, el alcance y adaptabilidad, y otros aspectos relevantes para el desarrollo de esta propuesta de herramienta.

3.2. Interfaz de usuario

Como se mencionó inicialmente, la propuesta de herramienta posibilita a que usuarios sin conocimiento en programación puedan usar este tipo de herramientas, por ello el

desarrollo de una interfaz de usuario intuitiva fue necesaria para lograr este cometido (ver Figura 9).

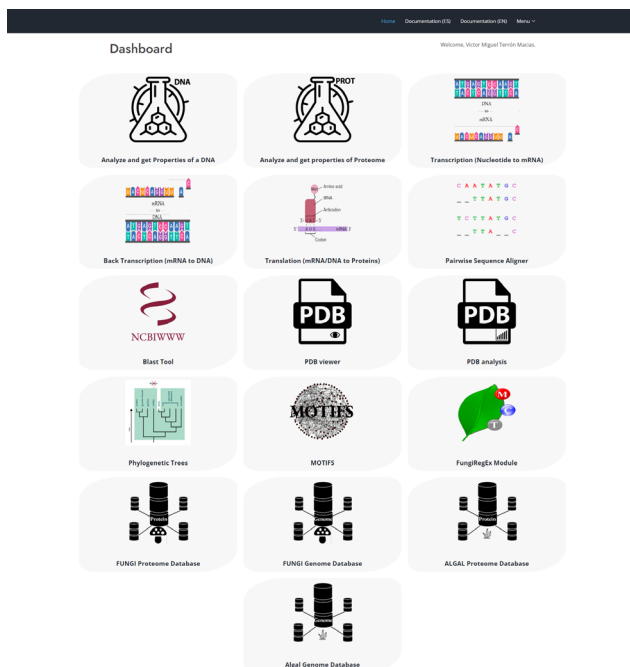


Figura 9 – Interfaz de usuario de la propuesta de herramienta.

La Figura 9 muestra la interfaz de usuario de la propuesta de herramienta con sus herramientas que están integradas dentro de la propuesta de aplicación.

### 3.3. Alcance y adaptabilidad

Esta propuesta de herramienta está diseñada para analizar secuencias biológicas de organismos de hongos y algas. Sin embargo, el software se adapta a una gama más amplia de secuencias biológicas sin importar si es o no de este tipo de organismos, lo que destaca su versatilidad. La adaptabilidad de la herramienta garantiza su utilidad en diversos contextos de investigación más allá del alcance inicial.

El tipo de secuencias que pueden analizar con la propuesta son secuencias genómicas y proteómicas de cualquier organismo, las secuencias genómicas son secuencias de nucleótidos o bases en una molécula de ADN (Brown, 2002). Por otra parte, las secuencias proteómicas son el total de moléculas proteicas presentes en una célula, tejido u órgano (Maino, 2000).

El alfabeto de las secuencias genómicas está conformado por 4 elementos: A, C, G y T; mientras que el alfabeto de las secuencias proteómicas se conforma por 26 elementos.

3.4.Optimización del flujo de trabajo

La integración de diferentes componentes de software en un mismo entorno es un punto clave del desarrollo de esta herramienta, ya que esta integración agiliza el flujo de trabajo eliminando la necesidad del usuario de navegar entre diferentes aplicaciones o software diferentes para el uso de sus herramientas o de tener conocimientos de programación.

3.5.Entrenamiento del usuario

Se ha preparado la documentación de esta propuesta de herramienta para facilitar su uso, con el objetivo de dotar al usuario del conocimiento necesario para el uso apropiado de la propuesta.

4. Resultados

La propuesta de herramienta presentó desafíos en la integración de tecnologías y componentes dentro de un mismo entorno principalmente por la diferencia entre los lenguajes de programación. Sin embargo, gracias al uso de diferentes tecnologías se logró una adecuada integración. A continuación, se presentan los resultados obtenidos, destacando cómo el uso de estas librerías ha contribuido a desarrollar la funcionalidad integral del software.

4.1. Integración de funcionalidades

La combinación de diferentes librerías en el desarrollo de la herramienta ha permitido la integración fluida de una amplia gama de funcionalidades relacionadas con el análisis de secuencias biológicas. Desde el análisis de propiedades de secuencias de ADN y proteínas hasta la generación de árboles filogenéticos, cada herramienta se ha implementado de manera coherente dentro del entorno unificado del software.

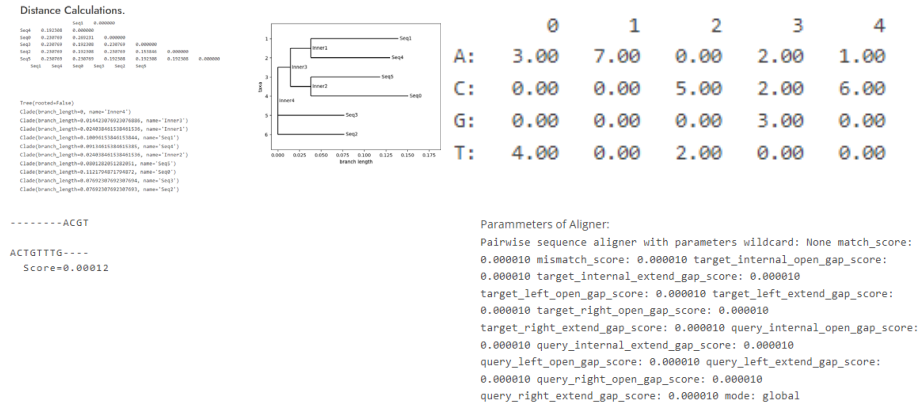


Figura 10 – Resultados cuantitativos.

La herramienta provee de resultados cuantitativos de los resultados que entrega, por ejemplo utilizando el generador de árboles filogenéticos muestra el cálculo de las

distancias entre las secuencias alineadas (ver Figura 10 esquina superior izquierda). Otro resultado cuantitativo que presenta la plataforma es al usar el alineador de secuencias por pares, ya que presenta el puntaje de alineamiento y una sección de estadísticas del alineador (ver Figura 10 sección inferior). Otro ejemplo de datos cuantitativos es al usar la herramienta de MOTIFS, ya que nos entrega una matriz de conteo de proteínas (ver Figura 10 esquina superior derecha).

#### **4.2. Eficiencia y Precisión**

El uso de librerías especializadas ha garantizado la eficiencia y precisión en el análisis de las secuencias biológicas. Por ejemplo, la implementación de BLAST ha permitido realizar búsquedas rápidas y precisas en la base de datos de NCBI, identificando secuencias similares con alta fiabilidad, funcionalidad que ha sido validada en el artículo publicado por P. Cock (Cock, 2009) que aborda la funcionalidad de BioPython. Asimismo, la integración de algoritmos de alineación de secuencias ha facilitado la comparación y el análisis de similitudes entre secuencias de ADN y proteínas que al igual esta funcionalidad ha sido validada por expertos en el campo que desarrollaron cada componente que se integró a esta herramienta.

#### **4.3. Facilidad de uso**

Gracias a la utilización de librerías bien establecidas y documentadas, unido al framework de DJANGO con sus integraciones la herramienta desarrollada ofrece una interfaz amigable y fácil de usar para los usuarios, destacando que se hizo más accesible el uso de herramientas bioinformáticas sin la necesidad de tener conocimientos en programación. Las funciones se presentan de manera clara, accesible y bien documentadas, lo que permite a los usuarios realizar análisis mediante el uso de los componentes de la herramienta de manera intuitiva y eficiente (Ver Figura 9).

#### **4.4. Escalabilidad y adaptabilidad**

La arquitectura del sistema y la forma en la que fue desarrollada la aplicación garantiza su escalabilidad y adaptabilidad a futuras actualizaciones y ampliaciones de funcionalidades ya que cada componente es independiente uno del otro, por tanto se pueden desarrollar más módulos mismos que se pueden integrar fácilmente a la herramienta sin comprometer otros componentes o funcionalidades ya desarrolladas, de esa forma se garantiza una constante actualización y adaptabilidad a futuras tecnologías, que es otra de las problemáticas identificadas por los expertos ante la disponibilidad de muchas herramientas que no reciben actualización.

### **5. Conclusiones**

En esta artículo se destaca la diversidad de herramientas bioinformáticas disponibles que presentan tanto ventajas como desafíos significativos respaldados por la opinión de expertos en el campo. Si bien esta variedad ofrece una amplia gama de funcionalidades, también plantea problemas de eficiencia y efectividad para los profesionales, como la dificultad para encontrar software adecuado y la falta de unificación en las herramientas disponibles.

En respuesta a estas problemáticas identificadas, se propone una solución innovadora: el desarrollo de una aplicación web que unifica diversas funcionalidades para análisis de secuencias biológicas. Esta propuesta surge como una respuesta directa a la necesidad de simplificar y optimizar el proceso de análisis, ofreciendo a los usuarios un entorno integrado que abarca múltiples funcionalidades sin requerir conocimientos de programación.

La elección de tecnologías sólidas, como Python y el framework Django, junto con la integración de librerías especializadas como BioPython, BioPandas, 3Dmol.js, BLAST, entre otras, proporciona una base robusta para el desarrollo de la herramienta propuesta. Aunado a la elección de una arquitectura modular, misma que asegura la escalabilidad y adaptabilidad de la aplicación a futuras actualizaciones y ampliaciones de funcionalidades sin comprometer funcionalidades ya implementadas.

La interfaz de usuario intuitiva y la documentación exhaustiva son aspectos clave de la propuesta, ya que facilitan la adopción y el uso adecuado de la herramienta por parte de usuarios de diversos niveles de experiencia. La capacidad de proporcionar resultados cuantitativos, respaldada por la validación de la eficacia de la herramienta, fortalece su utilidad y confiabilidad en el ámbito científico.

Esta propuesta de herramienta representa un avance significativo en la simplificación y mejora de los procesos de análisis de secuencias biológicas en el campo de la bioinformática. Su implementación promete no solo mejorar la eficiencia y efectividad de los profesionales en este campo, sino también impulsar el progreso científico y tecnológico en áreas afines.

## 6. Trabajo Futuro

A pesar de los avances significativos logrados con nuestra propuesta de herramienta, existen áreas que requieren atención adicional y mejoras para maximizar su utilidad y efectividad en el campo de la bioinformática. En este sentido, se identifican varias líneas de trabajo futuro que podrían enriquecer y fortalecer aún más nuestra herramienta: a) incluir la integración de algoritmos avanzados de análisis de secuencias, herramientas de visualización de datos mejoradas y métodos de análisis predictivo-basados en aprendizaje automático; b) expandir la herramienta para abarcar una gama más amplia de organismos, desde la medicina hasta la agricultura y la conservación de la biodiversidad.

## Referencias

Armstrong, J., Hickey, G., Diekhans, M., Fiddes, I. T., Novak, A. M., Deran, A., Fang, Q., Xie, D., Feng, S., Stiller, J., Genereux, D., Johnson, J., Marinescu, V. D., Alföldi, J., Harris, R. S., Lindblad-Toh, K., Haussler, D., Karlsson, E., Jarvis, E. D., Zhang, G. & Paten, B. (2020). Progressive Cactus is a multiple-genome aligner for the thousand-genome era. *Nature*, 246-251. <https://doi.org/10.1038/s41586-020-2871-y>

- Azza, E., Ahmed, J. M. (2021). Design considerations for workflow management systems use in production genomics research and the clinic. *Applications in Plant Sciences*, 16-27. <https://doi.org/10.1002/aps3.11538>
- Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information. (2017). National Center for Biotechnology Information (NCBI).
- Brown, T. A. (2002). *Genomes 2*. Garland Science.
- Butterfield, A. (2004). PyEvolve: a toolkit for statistical modelling of molecular evolution. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-5-1>
- Cazier, J.-B. (2013). Is there too many bioinformatics tools? *Encuesta*. Universidad de Birmingham. Obtenido de [https://www.researchgate.net/post/Is\\_there\\_too\\_many\\_bioinformatics\\_tools2/5](https://www.researchgate.net/post/Is_there_too_many_bioinformatics_tools2/5)
- Cock, P. J. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 1422-1423. <https://doi.org/10.1093/bioinformatics/btp163>
- Cogent3 Team. (s.f.). Cogent3. Obtenido de <https://cogent3.org/>
- Emelianova, K., Riaño-Pachón, D. M., Jimenez, M. F. T. (2023). Making sense of complexity: Advances in bioinformatics for plant biology. *Applications in Plant Sciences*. <https://doi.org/10.1002/aps3.11538>
- Gasteiger, E. (2003). ExPASy: The proteomics server for in-depth protein knowledge and analysis. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkg563>
- JGI. (2022). About Us. DOE Joint Genome Institute. Obtenido de <https://jgi.doe.gov/about-us/>
- Maino, R. (2000). Proteómica. El desafío de la post-genómica. *Naturee*, 403, 601.
- Raschka, S. (2017). BioPandas: Working with molecular structures in pandas DataFrames. *The Open Journal*. <https://doi.org/10.21105/joss.00279>
- Rego, N. A. (2014). 3Dmol.js: molecular visualization with WebGL. *Bioinformatics*, 1322-1324. <https://doi.org/10.1093/bioinformatics/btu829>
- Rob Knight, P. M. (2007). PyCogent: a toolkit for making sense from sequence. *genome Biology*. <https://doi.org/10.1186/gb-2007-8-8-r171>
- Vasileiou, D., Karapiperis, C., Baltsavia, I., Chasapi, A., Ahrén, D., Janssen, P. J., Iliopoulos, I., Promponas, V. J., Enright, A. J., Ouzounis, C. A. (2023). CGG toolkit: Software components for a computational genomics. *PLOS Computational Biology*, 1-10. <https://doi.org/10.1371/journal.pcbi.1011498>

## **Critérios Editoriais**

A RISTI (Revista Ibérica de Sistemas e Tecnologias de Informação) é um periódico científico, que foca a investigação e a aplicação prática inovadora no domínio dos sistemas e tecnologias de informação.

O Conselho Editorial da RISTI incentiva potenciais autores a submeterem artigos originais e inovadores para avaliação pelo Conselho Científico.

A submissão de artigos para publicação na RISTI deve realizar-se de acordo com as chamadas de artigos e as instruções e normas disponibilizadas no sítio Web da revista (<http://www.risti.xyz/>).

Todos os artigos submetidos são avaliados por um conjunto de membros do Conselho Científico, não inferior a três elementos.

Em cada número da revista são publicados entre cinco a oito dos melhores artigos submetidos.

## **Criterios Editoriales**

La RISTI (Revista Ibérica de Sistemas y Tecnologías de la Información) es un periódico científico, centrado en la investigación y en la aplicación práctica innovadora en el dominio de los sistemas y tecnologías de la información.

El Consejo Editorial de la RISTI incentiva autores potenciales a enviar sus artículos originales e innovadores para evaluación por el Consejo Científico.

El envío de artículos para publicación en la RISTI debe hacerse de conformidad con las llamadas de los artículos y las instrucciones y normas establecidas en el sitio Web de la revista (<http://www.risti.xyz/>).

Todos los trabajos enviados son evaluados por un número de miembros del Consejo Científico de no menos de tres elementos.

En cada número de la revista se publican cinco a ocho de los mejores artículos enviados.



Revista Ibérica de Sistemas e Tecnologias de Informação  
Revista Ibérica de Sistemas y Tecnologías de Información

©RISTI 2024 <http://www.risti.xyz>

